# ADTRAN OPERATING SYSTEM (AOS)

## Command Reference Guide

## AOS Version 13.1

July 2006
61950860L1-35N

## Trademarks

Any brand names and product names included in this manual are trademarks, registered trademarks, service marks, or trade names of their respective holders.

## To the Holder of this Manual

The contents of this manual are current as of the date of publication. ADTRAN reserves the right to change the contents without prior notice.

In no event will ADTRAN be liable for any special, incidental, or consequential damages or for commercial losses even if ADTRAN has been advised thereof as a result of issue of this publication.

## Software Licensing Agreement

Each ADTRAN product contains a single license for ADTRAN supplied software. Pursuant to the Licensing Agreement, you may: (a) use the software on the purchased ADTRAN device only and (b) keep a copy of the software for backup purposes. This Agreement covers all software installed on the system as well as any software available on the ADTRAN website. In addition, certain ADTRAN systems may contain additional conditions for obtaining software upgrades.

## Conventions


*Notes provide additional useful information.*


*Cautions signify information that could prevent service interruption.*

**WARNING** *Warnings provide information that could prevent damage to the equipment or endangerment to human life.*

901 Explorer Boulevard
P.O. Box 140000
Huntsville, AL 35814-4000
Phone: (256) 963-8000
www.adtran.com

## Warranty

ADTRAN will repair and return this product within the warranty period if it does not meet its published specifications or fails while in service. Warranty information can be found in the *Support* section of the ADTRAN website at http://www.adtran.com.

## Product Registration

Registering your product helps ensure complete customer satisfaction. Please take time to register your products in the *Support* section of the ADTRAN website at http://www.adtran.com

## Product Support Information

A return material authorization (RMA) is required prior to returning equipment to ADTRAN. For service, RMA requests, training, or more information, use the contact information shown below.

### Repair and Return

If you determine that a repair is needed, please contact our Customer and Product Service (CaPS) department to have an RMA number issued. CaPS should also be contacted to obtain information regarding equipment currently in house or possible fees associated with repair.

CaPS Department (256) 963-8722

Identify the RMA number clearly on the package (below the address), and return to the following address:

ADTRAN Customer and Product Service
901 Explorer Blvd. (East Tower)
Huntsville, Alabama 35806

RMA # _____

### Pre-Sale Inquiries and Applications Support

Your reseller should serve as the first point of contact for support. If additional pre-sales support is needed, the ADTRAN Support website provides a variety of support services such as a searchable knowledge base, the latest product documentation, application briefs, case studies, and a link to submit a question to an Applications Engineer. All of this, and more, is available in the *Support* section of the ADTRAN website at http://www.adtran.com.

When needed, further pre-sales assistance is available by calling our Applications Engineering Department.

Applications Engineering (800) 615-1176

### Post-Sale Support

Your reseller should serve as the first point of contact for support. If additional support is needed, the ADTRAN website provides a variety of support services such as a searchable knowledge base, updated firmware releases, latest product documentation, service request ticket generation and trouble-shooting tools. All of this, and more, is available in the *Support* section of the ADTRAN website at http://www.adtran.com.

When needed, further post-sales assistance is available by calling our Technical Support Center. Please have your unit serial number available when you call.

| | |
|---|---|
| Technical Support | (888) 4ADTRAN |
| International Technical Support | 1-256-963-8716 |

### Installation and Maintenance Support

The ADTRAN Custom Extended Services (ACES) program offers multiple types and levels of installation and maintenance services which allow you to choose the kind of assistance you need. This support is available at:

http://www.adtran.com/aces

For questions, call the ACES Help Desk.

| | |
|---|---|
| ACES Help Desk | (888) 874-ACES (2237) |

### Training

The Enterprise Network (EN) Technical Training Department offers training on our most popular products. These courses include overviews on product features and functions while covering applications of ADTRAN's product lines. ADTRAN provides a variety of training options, including customized training and courses taught at our facilities or at your site. For more information about training, please contact your Territory Manager or the Enterprise Training Coordinator.

| | |
|---|---|
| Training Phone | (800) 615-1176, ext. 7500 |
| Training Fax | (256) 963-6700 |
| Training Email | training@adtran.com |

## Export Statement

An Export License is required if an ADTRAN product is sold to a Government Entity outside of the EU+8 (Austria, Australia, Belgium, Czech Republic, Denmark, Finland, France, Germany, Greece, Hungary, Ireland, Italy, Japan, Luxembourg, Netherlands, New Zealand, Norway, Poland, Portugal, Spain, Sweden, Switzerland and the United Kingdom). This requirement is per DOC/BIS ruling G030477 issued 6/6/03. This product also requires that the Exporter of Record file a semi-annual report with the BXA detailing the information per EAR 740.17(5)(e)(2).

DOC - Department of Commerce
BIS - Bureau of Industry and Security
BXA - Bureau of Export Administration

# Table of Contents



# REFERENCE GUIDE INTRODUCTION

This manual provides information about the commands that are available with all of the NetVanta Series units and the Total Access 900 Series units.

If you are new to the ADTRAN Operating System's (AOS) Command Line Interface (CLI), take a few moments to review the information provided in the section which follows (<ItalicRef>CLI Introduction).

If you are already familiar with the CLI and you need information on a specific command or group of commands, proceed to *Command Descriptions* on page 14 of this guide.

# CLI INTRODUCTION

This portion of the Command Reference Guide is designed to introduce you to the basic concepts and strategies associated with using the AOS CLI.



## Accessing the CLI from your PC

All products using the AOS are initially accessed by connecting a VT100 terminal (or terminal emulator) to the **CONSOLE** port located on the rear panel of the unit using a standard DB-9 (male) to DB-9 (female) serial cable. Configure the VT100 terminal or terminal emulation software to the following settings:

- 9600 baud
- 8 data bits
- No parity
- 1 stop bit
- No flow control


*For more details on connecting to your unit, refer to the Quick Configuration Guides and Quick Start Guides located on the ADTRAN OS System Documentation CD provided with your unit.*

## Understanding Command Security Levels

The ADTRAN CLI has two command security levels — **Basic** and **Enable**. Both levels support a specific set of commands. For example, all interface configuration commands are accessible only through the Enable security level. The following table contains a brief description of each level.

| Level | Access by... | Prompt | With this level you can... |
|---|---|---|---|
| Basic | beginning an AOS session. | > | • display system information<br>• perform traceroute and ping functions<br>• open a Telnet session |
| Enable | entering **enable** while in the Basic command security level as follows:<br>>**enable** | # | • manage the startup and running configurations<br>• use the debug commands<br>• enter any of the configuration modes |


*To prevent unauthorized users from accessing the configuration functions of your AOS product, immediately install an Enable-level password. Refer to the Quick Configuration Guides and Quick Start Guides located on the ADTRAN OS System Documentation CD provided with your unit for more information on configuring a password.*

## Understanding Configuration Modes

The ADTRAN CLI has four configuration modes to organize the configuration commands – Global, Line, Router, and Interface. Each configuration mode supports a set of commands specific to the configurable parameters for the mode. For example, all Frame Relay configuration commands are accessible only through the interface configuration mode (for the virtual Frame Relay interface). The following table contains a brief description of each level.

| Mode | Access by... | Sample Prompt | With this mode you can... |
|---|---|---|---|
| Global | Entering **config** while at the Enable command security level prompt.<br>For example:<br>**>enable**<br>**#config term** | (config)# | • set the system's Enable-level password(s)<br>• configure the system global IP parameters<br>• configure the SNMP parameters<br>• enter any of the other configuration modes |
| Line | Specifying a line (console or Telnet) while at the Global Configuration mode prompt.<br>For example:<br>**>enable**<br>**#**config term<br>(config)**#line console 0** | (config-con0)# | • configure the console terminal settings (datarate, login password, etc.)<br>• create Telnet logins and specify their parameters (login password, etc.) |
| Router | Entering **router rip** or **router ospf** while at the Global Configuration mode prompt.<br>For example:<br>**>enable**<br>**#config term**<br>(config)**#router rip** | (config-rip)# | • configure RIP or OSPF parameters<br>• suppress route updates.<br>• redistribute information from outside routing sources (protocols) |
| Interface | Specifying an interface (T1, Ethernet, Frame Relay, ppp, etc.) while in the Global Configuration mode.<br>For example:<br>**>enable**<br>**#config term**<br>(config)**#interface eth 0/1** | (config-eth 0/1)#<br><br>(The above prompt is for the Ethernet **LAN** interface located on the rear panel of the unit.) | • configure parameters for the available LAN and WAN interfaces |

## Using CLI Shortcuts

The ADTRAN CLI provides several shortcuts which help you configure your AOS product more easily. See the following table for descriptions.

| Shortcut | Description |
|---|---|
| Up arrow key | To re-display a previously entered command, use the up arrow key. Continuing to press the up arrow key cycles through all commands entered starting with the most recent command. |
| **<Tab>** key | Pressing the **<Tab>** key after entering a partial (but unique) command will complete the command, display it on the command prompt line, and wait for further input. |
| **?** | The ADTRAN CLI contains help to guide you through the configuration process. Using the question mark, do any of the following:<br>Display a list of all subcommands in the current mode. For example:<br><br>(config-t1 1/1)**#coding ?**<br>ami - Alternate Mark Inversion<br>b8zs - Bipolar Eight Zero Substitution<br><br>Display a list of available commands beginning with certain letter(s). For example:<br><br>(config)**#ip d?**<br>default-gateway dhcp-server domain-lookup domain-name domain-proxy<br><br>Obtain syntax help for a specific command by entering the command, a space, and then a question mark (?). The ADTRAN CLI displays the range of values and a brief description of the next parameter expected for that particular command. For example:<br><br>(config-eth 0/1)**#mtu ?**<br><64-1500> - MTU (bytes) |
| **<Ctrl + A>** | Jump to the beginning of the displayed command line. This shortcut is helpful when using the **no** form of commands (when available). For example, pressing **<Ctrl + A>** at the following prompt will place the cursor directly after the **#**:<br>(config-eth 0/1)**#ip address 192.33.55.6** |
| **<Ctrl + E>** | Jump to the end of the displayed command line. For example, pressing **<Ctrl + E>** at the following prompt will place the cursor directly after the **6**:<br><br>(config-eth 0/1)**#ip address 192.33.55.6** |
| **<Ctrl + U>** | Clears the current displayed command line. The following provides an example of the **<Ctrl + U>** feature:<br><br>(config-eth 0/1)**#ip address 192.33.55.6** (Press **<Ctrl + U>** here)<br>(config-eth 0/1)# |
| auto finish | You need only enter enough letters to identify a command as unique. For example, entering **int t1 1/1** at the Global configuration prompt provides you access to the configuration parameters for the specified T1 interface. Entering **interface t1 1/1** would work as well, but is not necessary. |

## Performing Common CLI Functions

The following table contains descriptions of common CLI commands.

| Command | Description |
|---|---|
| **do** | The **do** command provides a way to execute commands in other command sets without taking the time to exit the current and enter the desired one. The following example shows the **do** command used to view the Frame Relay interface configuration while currently in the T1 interface command set:<br><br>(config)**#interface t1 1/1**<br>(config-t1 1/1)**#do show interfaces fr 7** |
| **no** | To undo an issued command or to disable a feature, enter **no** before the command.<br><br>For example:<br>**no shutdown t1 1/1** |
| **copy running-config startup-config** | When you are ready to save the changes made to the configuration, enter this command. This copies your changes to the unit's nonvolatile random access memory (NVRAM). Once the save is complete, the changes are retained even if the unit is shut down or suffers a power outage. |
| **show running config** | Displays the current configuration. |
| **debug** | Use the **debug** command to troubleshoot problems you may be experiencing on your network. These commands provide additional information to help you better interpret possible problems. For information on specific debug commands, refer to the section *Enable Mode Command Set* on page 37. |
| **undebug all** | To turn off any active debug commands, enter this command. |


*The overhead associated with the **debug** command takes up a large portion of your AOS product's resources and at times can halt other processes. It is best to only use the **debug** command during times when the network resources are in low demand (non-peak hours, weekends, etc.).*

## Understanding CLI Error Messages

The following table lists and defines some of the more common error messages given in the CLI.

| Message | Helpful Hints |
|---|---|
| **%Ambiguous command<br>%Unrecognized Command** | The command may not be valid in the current command mode, or you may not have entered enough correct characters for the command to be recognized. Try using the **?** command to determine your error. Refer to *Using CLI Shortcuts* on page 11 for more information. |
| **%Invalid or incomplete command** | The command may not be valid in the current command mode, or you may not have entered all of the pertinent information required to make the command valid. Try using the **?** command to determine your error. Refer to *Using CLI Shortcuts* on page 11 for more information. |
| **%Invalid input detected at "^" marker** | The error in command entry is located where the caret (^) mark appears. Enter a question mark at the prompt. The system will display a list of applicable commands or will give syntax information for the entry. |

## COMMAND DESCRIPTIONS

This portion of the guide provides a detailed listing of all available commands for the ADTRAN CLI (organized by command set). Each command listing contains pertinent information including the default value, a description of all sub-command parameters, functional notes for using the command, and a brief technology review. To search for a particular command alphabetically, use the Index at the end of this document. To search for information on a group of commands within a particular command set, use the linked references given below:



**Line Interface Command Sets**



**Physical Interface Command Sets**



**Virtual Interface Command Sets**



**VPN Parameter Command Sets**



**Routing Command Sets**



**Voice Command Sets**



**Security and Services Command Sets**



*TACACS+ Group Configuration Command Set* on page 2027

# BASIC MODE COMMAND SET

To activate the Basic mode, simply log in to the unit. After connecting the unit to a VT100 terminal (or terminal emulator) and activating a terminal session, the following prompt displays:

**>**

The following command is common to multiple command sets and is covered in a centralized section of this guide. For more information, refer to the section listed below:

*exit on page 35*

All other commands for this command set are described in this section in alphabetical order.



# enable

Use the **enable** command (at the Basic Command mode prompt) to enter the Enable Command mode. Use the **disable** command to exit the Enable Command mode. Refer to *Enable Mode Command Set* on page 37 for more information.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

The Enable Command mode provides access to operating and configuration parameters and should be password protected to prevent unauthorized use. Use the **enable password** command (found in the Global Configuration mode) to specify an Enable Command mode password. If the password is set, access to the Enable Commands (and all other “privileged” commands) is only granted when the correct password is entered. Refer to *enable password <password>* on page 479 for more information.

## Usage Examples

The following example enters the Enable Command mode and defines an Enable Command mode password:

```
>enable
#configure terminal
(config)#enable password ADTRAN
```

At the next login, the following sequence must occur:

```
>enable
Password: ******
#
```

# logout

Use the **logout** command to terminate the current session and return to the login screen.

## Syntax Description

No subcommands.

## Default Values

No defaults necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example shows the logout command being executed in the Basic mode:

>**logout**

Session now available

Press RETURN to get started.

# ping *<ip address>*

Use the **ping** command (at the Basic Command mode prompt) to verify Internet Protocol (IP) network connectivity.Variations of this command include:

**ping**
**ping** <*ip address*>
**ping** <*ip address*> **data** <*string*>
**ping** <*ip address*> **repeat** <*number*>
**ping** <*ip address*> **size** <*value*>
**ping** <*ip address*> **source** <*ip address*>
**ping** <*ip address*> **timeout** <*value*>

## Syntax Description

| | |
|---|---|
| <*ip address*> | Optional. Specifies the IP address of the system to ping. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**).<br>Entering the **ping** command with no specified IP address prompts the user with parameters for a more detailed **ping** configuration. Refer to *Functional Notes* (below) for more information. |
| **data** <*string*> | Specifies an alphanumerical string to use (the ASCII equivalent) as the data pattern in the ECHO_REQ packets. |
| **repeat** <*number*> | Specifies the number of ping packets to send to the system. Valid range is 1 to 1,000,000. |
| **size** <*value*> | Specifies the datagram size (in bytes) of the ping packet. Valid range is 1 to 1448 bytes. |
| **source** <*ip address*> | Specifies the IP address to use as the source address in the ECHO_REQ (or interface) packets. |
| **timeout** <*value*> | Specifies the timeout period after which the ping is considered unsuccessful. Valid range is 1 to 5 seconds. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

The **ping** command helps diagnose basic IP network connectivity using the Packet Internet Groper program to repeatedly bounce Internet Control Message Protocol (ICMP) Echo_Request packets off a system (using a specified IP address). AOS allows executing a standard **ping** request to a specified IP address or provides a set of prompts to configure a more specific **ping** configuration.

## Usage Examples

The following is an example of a successful **ping** command:

>**ping**
Target IP address:**10.10.10.1**
Repeat count[1-1000000]:**5**
Datagram Size [1-1000000]:**100**
Timeout in seconds [1-5]:**2**
Extended Commands? [y or n]:**n**
Type CTRL+C to abort.
Legend: '!' = Success '?' = Unknown host '$' = Invalid host address
'*' = Request timed out '-' = Destination host unreachable
'x' = TTL expired in transit

Pinging 192.168.0.30 with 100 bytes of data:
!!!!!
Success rate is 100 percent (5/5) round-trip min/avg/max = 19/20.8/25 ms

# show clock

Use the **show clock** command to display the system time and date entered using the **clock set** command. Refer to the section *clock set* <*time*> <*day*> <*month*> <*year*> on page 90 for more information.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example displays the current time and data from the system clock:

```
>show clock
23:35:07 UTC Tue Aug 20 2002
```

# show snmp

Use the **show snmp** command to display the system Simple Network Management Protocol (SNMP) parameters and current status of SNMP communications.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following is an example output using the **show snmp** command for a system with SNMP disabled and the default chassis and contact parameters:

```
>show snmp
Chassis: Chassis ID
Contact: Customer Service
0 Rx SNMP packets
  0 Bad community names
  0 Bad community uses
  0 Bad versions
  0 Silent drops
  0 Proxy drops
  0 ASN parse errors
```

# show version

Use the **show version** command to display the current AOS version information.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following is a sample **show version** output:

>**show version**

```
AOS version 06.01.00
  Checksum: 1F0D5243 built on Fri Nov 08 13:12:06 2002
  Upgrade key: de76efcfeb4c8eeb6901188475dd0917
Boot ROM version 03.00.18
  Checksum: 7A3D built on: Fri Nov 08 13:12:25 2002
Copyright (c) 1999-2002 ADTRAN Inc.
Serial number C14C6308

UNIT_2 uptime is 0 days 4 hours 59 minutes 43 seconds

System returned to ROM by Warm Start
Current system image file is "030018adv.biz"
Boot system image file is “030018adv.biz”
```

# **telnet** *<ip address>*

Use the **telnet** command to open a Telnet session (through the AOS) to another system on the network.

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies the IP address of the remote system. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example opens a Telnet session with a remote system (**10.10.10.1**):

```
>telnet 10.10.10.1
User Access Login
Password:
```

# traceroute *<ip address>*

Use the **traceroute** command to display the Internet Protocol (IP) routes a packet takes to reach the specified destination.

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies the IP address of the remote system to trace the routes to. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example performs a traceroute on the IP address **192.168.0.1**:

```
#traceroute 192.168.0.1
Type CTRL+C to abort.
Tracing route to 192.168.0.1 over a maximum of 30 hops
  1    22ms   20ms   20ms    192.168.0.65
  2    23ms   20ms   20ms    192.168.0.1
#
```

# COMMON COMMANDS

The following section contains descriptions of commands that are common across multiple command sets. These commands are listed in alphabetical order.



# alias <*"text"*>

Use the **alias** command to populate the ifAlias OID (Interface Table MIB of RFC2863) for all physical and virtual interfaces when using Simple Network Management Protocol (SNMP) management stations. Use the **no** form of this command to remove an alias.

## Syntax Description

| | |
|---|---|
| <*"text"*> | Describes the interface (for SNMP) using an alphanumeric character string enclosed in quotation marks (limited to 64 characters). |

## Default Values

No defaults required for this command.

## Applicable Command Modes

Applies to all interface mode command sets.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

The ifAlias OID is a member of the ifXEntry object-type (defined in RFC2863) used to provide a non-volatile, unique name for various interfaces. This name is preserved through power cycles. Enter a string (using the **alias** command) which clearly identifies the interface.

## Usage Examples

The following example defines a unique character string for the T1 interface:

(config)**#interface t1 1/1**
(config-t1 1/1)**#alias "CIRCUIT_ID_23-908-8887-401"**

## Technology Review

Please refer to RFC2863 for more detailed information on the ifAlias display string.

# cross-connect

Use the **cross-connect** command to create a cross-connect map from a created TDM group on an interface to a virtual interface. Variations of this command include:

**cross-connect** *<number> <from interface> <to interface>*
**cross-connect** *<number> <from interface> <group number> <to interface>*


*Changing **cross-connect** settings could potentially result in service interruption.*

## Syntax Description

| | |
|---|---|
| *<number>* | Identifies the cross-connect using a number descriptor or label for (useful in systems that allow multiple cross-connects). |
| *<from interface>* | Specifies the interface (physical or virtual) on one end of the cross-connect. Specify an interface in the format *<interface type [slot/port \| slot/port.sub-interface id \| interface id \| interface id.sub-interface id]>*. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Enter **cross-connect 1 ?** for a list of valid interfaces. |
| *<group number>* | Optional. Specifies which configured TDM group to use for this cross-connect. This subcommand only applies to T1 physical interfaces. |
| *<to interface>* | Specifies the virtual interface on the other end of the cross-connect. Specify an interface in the format *<interface type [slot/port \| slot/port.sub-interface id \| interface id \| interface id.sub-interface id]>*. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Use the **?** to display a list of valid interfaces. |

## Default Values

By default, there are no configured cross-connects.

## Applicable Command Modes

Applies to all interface mode command sets.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include the E1 interface. |

## Functional Notes

Cross-connects provide the mechanism for connecting a configured virtual (layer 2) endpoint with a physical (layer 1) interface. Supported layer 2 protocols include Frame Relay and point-to-point protocol (PPP).

## Usage Examples

The following example creates a Frame Relay endpoint and connects it to the T1 1/1 physical interface:

1. Create the Frame Relay virtual endpoint and set the signaling method:

(config)**#interface frame-relay 1**
(config-fr 1)**#frame-relay lmi-type cisco**

2. Create the sub-interface and configure the PVC parameters (including DLCI and IP address):

(config-fr 1)**#interface fr 1.1**
(config-fr 1.1)**#frame-relay interface-dlci 17**
(config-fr 1.1)**#ip address 168.125.33.252 255.255.255.252**

3. Create the TDM group of 12 DS0s (64K) on the T1 physical interface:

(THIS STEP IS ONLY VALID FOR T1 INTERFACES.)

(config)**#interface t1 1/1**
(config-t1 1/1)**#tdm-group 1 timeslots 1-12 speed 64**
(config-t1 1/1)**#exit**

4. Connect the Frame Relay sub-interface with port T1 1/1:

(config)**#cross-connect 1 t1 1/1 1 fr 1**

## Technology Review

Creating an endpoint that uses a layer 2 protocol (such as Frame Relay) is generally a four-step process:

Step 1:
Create the Frame Relay virtual endpoint (using the **interface frame-relay** command) and set the signaling method (using the **frame-relay lmi-type** command). Also included in the Frame Relay virtual endpoint are all the applicable Frame Relay timers logging thresholds, encapsulation types, etc. Generally, most Frame Relay virtual interface parameters should be left at their default state. For example, the following creates a Frame Relay interface labeled **7** and sets the signaling method to **ansi**.

(config)**#interface frame-relay 7**
(config-fr 7)**#frame-relay lmi-type ansi**

Step 2:
Create the sub-interface and configure the PVC parameters. Using the sub-interface, apply access policies to the interface, create bridging interfaces, configure dial-backup, assign an IP address, and set the PVC data-link control identifier (DLCI). For example, the following creates a Frame Relay sub-interface labeled **22,** sets the DLCI to **30,** and assigns an IP address of **193.44.69.253** to the interface.

(config-fr 7)**#interface fr 7.22**
(config-fr 7.22)**#frame-relay interface-dlci 30**
(config-fr 7.22)**#ip address 193.44.69.253 255.255.255.252**

Step 3: (VALID ONLY FOR T1 INTERFACES)
Specify the group of DS0s used for signaling on the T1 interface by creating a TDM group. Group any number of contiguous DS0s together to create a data pipe for layer 2 signaling. Also use the **tdm-group** command to specify the per-DS0 signaling rate on the interface. For example, the following creates a TDM group labeled **9** containing 20 DS0s (each DS0 having a data rate of 56 kbps).

(config)**#interface t1 1/1**
(config-t1 1/1)**#tdm-group 9 timeslots 1-20 speed 56**
(config-t1 1/1)**#exit**

Step 4:
Make the association between the layer 2 endpoint and the physical interface using the **cross-connect** command. Supported layer 2 protocols include Frame Relay and point-to-point protocol (PPP). For example, the following creates a cross-connect (labeled **5**) to make an association between the Frame Relay virtual interface (**fr 7**) and the TDM group configured on interface t1 1/1 (**tdm-group 9**).

(config)**#cross-connect 5 t1 1/1 9 fr 7**

# description *<text>*

Use the **description** command to identify the specified interface (for example, circuit ID, contact information, etc.). Use the **no** form of this command to remove a description.

## Syntax Description

| | |
|---|---|
| *<text>* | Identifies the specified interface using up to 80 alphanumeric characters. |

## Default Values

No defaults required for this command.

## Applicable Command Modes

Applies to all interface mode command sets.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example enters comment information using the **description** command:

(config)**#interface t1 1/1**
(config-t1 1/1)**#description This is the Dallas office T1**

# do

Use the **do** command to execute any AOS command, regardless of the active configuration mode. It provides a way to execute commands in other modes without taking the time to exit the current mode and enter the desired one.

## Syntax Description

No subcommands.

## Default Values

No defaults required for this command.

## Applicable Command Modes

Applies to all mode command sets.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |

## Functional Notes

Use the **do** command to view configurations or interface states after configuration changes are made without exiting to the Enable mode.

## Usage Examples

The following example shows the **do** command used to view the Frame Relay interface configuration while currently in the T1 Interface Configuration mode:

```
(config)#interface t1 1/1
(config-t1 1/1)#do show interfaces fr 7
fr 7 is ACTIVE
  Signaling type is ANSI signaling role is USER
  Polling interval is 10 seconds full inquiry interval is 6 polling intervals
Output queue: 0/0 (highest/drops)
    0 packets input 0 bytes
    0 pkts discarded 0 error pkts 0 unknown protocol pkts
    0 packets output 0 bytes
    0 tx pkts discarded 0 tx error pkts
```

# end

Use the **end** command to exit the current configuration mode and enter the Enable Security mode.

*When exiting the Global Configuration mode, remember to perform a **copy running-config startup-config** to save all configuration changes.*

## Syntax Description

No subcommands.

## Default Values

No defaults necessary for this command.

## Applicable Command Modes

Applies to all mode command sets except Basic mode.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example shows the **end** command being executed in the T1 Interface Configuration mode:

```
(config-t1 1/1)#end
#
```

#- Enable Security mode command prompt

# exit

Use the **exit** command to exit the current configuration mode and enter the previous one. For example, using the **exit** command in an interface configuration mode will activate the Global Configuration mode. When using the **exit** command in the Basic mode, the current session will be terminated.


*When exiting the Global Configuration mode, remember to perform a **copy running-config startup-config** to save all configuration changes.*

## Syntax Description

No subcommands.

## Default Values

No defaults necessary for this command.

## Applicable Command Modes

Applies to all mode command sets.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example shows the **exit** command being executed in the Global Configuration mode:

```
(config)#exit
#
```

**#**- Enable Security mode command prompt

# shutdown

Use the **shutdown** command to disable the interface (both physical and virtual) so that no data will be passed through. Use the **no** form of this command to turn on the interface and allow it to pass data. By default, all interfaces are disabled.

## Syntax Description

No subcommands.

## Default Values

By default, all interfaces are disabled.

## Applicable Command Modes

Applies to all interface mode command sets.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example administratively disables the modem interface:

(config)**#interface modem 1/2**
(config-modem 1/2)**#shutdown**

# ENABLE MODE COMMAND SET

To activate the Enable mode, enter the **enable** command at the Basic mode prompt. (If an enable password has been configured, a password prompt will display.) For example:

```
>enable
Password: XXXXXXX
#
```

The following command is common to multiple command sets and is covered in a centralized section of this guide. For more information, refer to the section listed below:

*exit on page 35*

All other commands for this command set are described in this section in alphabetical order.



# clear access-list *<name>*

Use the **clear access-list** command to clear all counters associated with all access control lists (ACLs) or a specified ACL.

## Syntax Description

| | |
|---|---|
| *<name>* | Optional. Specifies the name (label) of an ACL. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example clears all counters for the access control list labeled **MatchAll**:

```
>enable
#clear access-list MatchAll
```

# clear arp-cache

Use the **clear arp-cache** command to remove all dynamic entries from the Address Resolution Protocol (ARP) cache table.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example removes all dynamic entries from the ARP cache:

```
>enable
#clear arp-cache
```

# clear arp-entry *<ip address>*

Use the **clear arp-entry** command to remove a single entry from the Address Resolution Protocol (ARP) cache.

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies a valid IP address to remove. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example removes the entry for 10.10.10.1 from the ARP cache:

**>enable**
**#clear arp-entry 10.10.10.1**

# clear bridge <*number*>

Use the **clear bridge** command to clear all counters associated with bridging (or for a specified bridge group). Variations of this command include:

**clear bridge**
**clear bridge** <*number*>

## Syntax Description

| | |
|---|---|
| <*number*> | Optional. Specifies a single bridge group. Range is 1 to 255. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example clears all counters for bridge group 17:

**>enable**
**#clear bridge 17**

# clear buffers max-used

Use the **clear buffers max-used** command to clear the maximum-used statistics for buffers displayed in the **show memory heap** command.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Usage Examples

The following example clears the maximum-used buffer statics:

```
>enable
#clear buffers max-used
```

# clear counters *<interface>*

Use the **clear counters** command to clear all interface counters (or the counters for a specified interface).

## Syntax Description

| | |
|---|---|
| *<interface>* | Specifies an interface in the format *<interface type [slot/port \| slot/port.sub-interface id \| interface id \| interface id.sub-interface id]>*. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **clear counters ?** or **show interface ?** for a complete list of interfaces. |

## Default Values

No default values necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 9.1 | Command was expanded to include HDLC and tunnel interfaces. |

## Usage Examples

The following example clears all counters associated with the Ethernet 0/1 interface:

```
>enable
#clear counters ethernet 0/1
```

# clear counters media-gateway

Use the **clear counters media-gateway** command to reset cumulative totals for all Realtime Transport Protocol (RTP) channels or for a specific RTP channel. Variations of this command include the following:

**clear counters media-gateway**
**clear counters media-gateway channel** <*value*>

## Syntax Description

| | |
|---|---|
| **channel** <*value*> | Optional. Specifies the ID of a particular media-gateway channel to be reset (for example, 0/1.1). |

## Default Values

No default value is necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example resets the counters on media gateway **channel 0/1.1**:

>**enable**
**#clear counters media-gateway channel 0/1.1**
Counters on media-gateway channel reset by console.

# clear counters probe

Use the **clear counters probe** command to reset counters on all probe objects or on a specific probe. Variations of this command include:

**clear counters probe**
**clear counters probe** <*name*>

## Syntax Description

| | |
|---|---|
| <*name*> | Specifies a probe object to reset counter. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example resets the counters for all configured probes:

**>enable**
**#clear counters probe**

The following example resets the counters only for the probe named **probe_A**:

**>enable**
**#clear counters probe probe_A**

# clear counters track

Use the **clear counters track** command to reset counters on all track objects or on a specifically named track.

**clear counters track**
**clear counters track** <*name*>

## Syntax Description

| | |
|---|---|
| <*name*> | Specifies a track object to reset counter. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example resets the counters for all configured tracks:

**>enable**
**#clear counters track**

The following example resets the counters only for the track named **track_1**:

**>enable**
**#clear counters track track_1**

# clear counters vlan *<vlan id>*

Use the **clear counters vlan** command to reset counters on the specified virtual local area network (VLAN) interface.

## Syntax Description

| | |
|---|---|
| *<vlan id>* | Specifies a valid VLAN interface ID. Range is 1 to 4094. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following example resets the counters on VLAN interface 7:

```
>enable
#clear counters vlan 7
```

# clear counters voice-trunk

Use the **clear counters voice-trunk** command to reset counters on all voice trunks or on a specific voice trunk. Variations of this command include:

**clear counters voice-trunk all**
**clear counters voice-trunk** *<trunk id>*

## Syntax Description

| | |
|---|---|
| **all** | Clears all voice trunk counters. |
| *<trunk id>* | Specifies clearing a specific voice trunk using the trunk's 2-digit identifier following T (for example, T01). |

## Default Values

No default value is necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example resets the counters for all configured voice trunks:

>**enable**
**#clear counters voice-trunk all**

# clear crypto ike sa

Use the **clear crypto ike sa** command to clear existing IKE security associations (SAs). Use the **policy** and **remote-id** options to clear specific SAs without clearing them all. Variations of this command include:

**clear crypto ike sa**
**clear crypto ike sa policy** <*value*>
**clear crypto ike sa remote-id** <*remote id*>

## Syntax Description

| | |
|---|---|
| **policy** <*value*> | Optional. Removes all IKE SAs associated with the specified policy priority value. This number is assigned using the command *crypto ike* on page 467. |
| **remote-id** <*remote id*> | Optional. Removes all IKE SAs associated with the specified IKE remote ID. A delete payload is sent to the peers prior to deletion of the SA. This command is preferred to the **clear crypto ike sa policy** <*value*> command when multiple unique SAs have been created on the same IKE policy but the user wants to delete only the SA to a unique peer. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 4.1 | Command was introduced. |
| Release 12.1 | Command was expanded to include **policy** and **remote-id**. |

## Usage Examples

The following example clears the entire database of IKE SAs (including the active associations):

**>enable**
**#clear crypto ike sa**

The following example clears IKE SAs associated with **policy 101**:

**>enable**
**#clear crypto ike sa policy 101**

The following example clears an IKE SA associated with **remote-id netvanta**:

**>enable**
**#clear crypto ike sa remote-id netvanta**

# clear crypto ipsec sa

Use the **clear crypto ipsec sa** command to clear existing IPSec security associations (SAs), including active ones. Variations of this command include the following:

**clear crypto ipsec sa**
**clear crypto ipsec sa entry** <*ip address*> **ah** <*SPI*>
**clear crypto ipsec sa entry** <*ip address*> **esp** <*SPI*>
**clear crypto ipsec sa map** <*name*>
**clear crypto ipsec sa peer** <*ip address*>

## Syntax Description

| | |
|---|---|
| **entry** <*ip address*> | Optional. Clears only the SAs related to the specified destination IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| **ah** <*SPI*> | Optional. Clears only a portion of the SAs by specifying the authentication header (AH) protocol and a security parameter index (SPI). You can determine the correct SPI value using the **show crypto ipsec sa** command. |
| **esp** <*SPI*> | Optional. Clears only a portion of the SAs by specifying the encapsulating security payload (ESP) protocol and an SPI. You can determine the correct SPI value using the **show crypto ipsec sa** command. |
| **map** <*name*> | Optional. Clears only the SAs associated with the specified crypto map. |
| **peer** <*ip address*> | Optional. Clears only the SAs associated with the specified far-end IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 4.1 | Command was introduced. |

## Usage Examples

The following example clears all IPSec SAs:

> **enable**
**#clear crypto ipsec sa**

The following example clears the IPSec SA used for ESP traffic with the SPI of 300 to IP address 63.97.45.57:

> **enable**
**#clear crypto ipsec sa entry 63.97.45.57 esp 300**

# clear dump-core

The **clear dump-core** command clears diagnostic information appended to the output of the **show version** command. This information results from an unexpected unit reboot.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Usage Examples

The following example clears the entire database of IKE SAs (including the active associations):

```
>enable
#clear dump-core
```

# clear event-history

Use the **clear event-history** command to clear all messages logged to the local event-history.


*Messages cleared from the local event-history (using the **clear event-history** command) are no longer accessible.*

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example clears all local event-history messages:

```
>enable
#clear event-history
```

# clear gvrp statistics

Use the **clear gvrp statistics** command to clear counter statistics on GARP VLAN Registration Protocol (GVRP) interfaces. Variations of this command include:

**clear gvrp statistics all**
**clear gvrp statistics interface** <*interface*>

## Syntax Description

| | |
|---|---|
| **all** | Clears the information for all GVRP interfaces. |
| **interface** <*interface*> | Clears the information for the specified interface. Specify an interface in the format <*interface type [slot/port* \| *slot/port.sub-interface id* \| *interface id* \| *interface id.sub-interface id]*>. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **clear gvrp statistics interface ?** for a complete list of applicable interfaces. |

## Default Values

There are no default settings for this command.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Usage Examples

The following example clears counter statistics on the GVRP interfaces:

>**enable**
**#clear gvrp statistics all**

# clear host

Use the **clear host** command to clear a host name when using the Domain Naming System (DNS) proxy. Variations of this command include:

**clear host ***
**clear host** <*name*>

## Syntax Description

| | |
|---|---|
| * | Clears all hosts from the host table. |
| <*name*> | Clears a specific host entry from the host-to-address table. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example clears all dynamic host names:

>**enable**
**#clear host ***

# clear ip bgp

Use the **clear ip bgp** command to clear BGP neighbors as specified. Variations of this command include:

**clear ip bgp [* |** *<number>* **|** *<ip address>***] in**
**clear ip bgp [* |** *<number>* **|** *<ip address>***] out**
**clear ip bgp [* |** *<number>* **|** *<ip address>***] soft**

## Syntax Description

| | |
|---|---|
| ***** | Clears all BGP neighbors. |
| *<number>* | Clears all BGP neighbors with the specified autonomous system (AS) number. Range is 1 to 65,535. |
| *<ip address>* | Clears the BGP neighbor with the specified IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| **in** | Causes a *soft* reset inbound with a neighbor, reprocessing routes advertised by that neighbor. |
| **out** | Causes a *soft* reset outbound with a neighbor, re-sending advertised routes to that neighbor. |
| **soft** | Causes a **soft** reset both inbound and outbound. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Functional Notes

The **clear ip bgp** command must be issued to re-initialize the BGP process between the peers matching the given arguments. Most neighbor changes, including changes to prefix-list filters, do not take effect until the **clear** command is issued. A hard reset clears the TCP connection with the specified peers, which results in clearing the table. This method of clearing is disruptive and causes peer routers to record a route flap for each route.

The **out** version of this command provides a soft reset out to occur by causing all routes to be re-sent to the specified peer(s). TCP connections are not torn down, so this method is less disruptive. Output filters/policies are re-applied before sending the update.

The **in** version of this command provides a soft reset in to occur by allowing the router to receive an updated table from a peer without tearing down the TCP connection. This method is less disruptive and does not count as a route flap. Currently, all of the peer's routes are stored permanently, even if they are filtered by a prefix list. The command causes the peer's routes to be reprocessed with any new parameters.

## Usage Examples

The following example causes a hard reset with peers with an AS number of 101:

>**enable**
**#clear ip bgp 101**

# clear ip cache

Use the **clear ip cache** command to delete cache table entries.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example removes all entries from the cache table:

```
>enable
#clear ip cache
```

# clear ip ffe

Use the **clear ip ffe** command to remove the FastFlow Engine (FFE) entries on all interfaces or on a specific ingress interface. Variations of this command include:

**clear ip ffe** <*interface*>

## Syntax Description

| | |
|---|---|
| <*interface*> | Specifies an interface in the format <*interface type [slot/port | slot/port.sub-interface id | interface id | interface id.sub-interface id]*>. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **clear ip ffe?** for a complete list of valid interfaces. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example clears all FFE entries for the Ethernet 0/1 interface:

>**enable**
**#clear ip ffe ethernet 0/1**

# clear ip igmp group

Use the **clear ip igmp group** command to clear entries from the Internet Group Management Protocol (IGMP) tables. If no address or interface is specified, all non-static IGMP groups are cleared with this command. Variations of this command include:

**clear ip igmp group**
**clear ip igmp group** *<multicast address>*
**clear ip igmp group** *<interface>*

## Syntax Description

| | |
|---|---|
| *<multicast address>* | Optional. Clears the IGMP tables of a specific multicast group IP address. The multicast group IP address range is 244.0.0.0 to 239.255.255.255 or 224.0.0.0 /4. |
| *<interface>* | Optional. Clears the IGMP tables of all interfaces of the specified type or a specific interface of a particular type. Specify an interface in the format *<interface type [slot/port \| slot/port.sub-interface id \| interface id \| interface id.sub-interface id]>*. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **clear ip igmp group ?** for a list of valid interfaces. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |
| Release 9.1 | Command was expanded to include HDSL and tunnel interfaces. |

## Usage Examples

The following example shows output for the **show igmp groups** command before and after a **clear ip igmp group** command is issued. This example clears the IGMP entry that was registered dynamically by a host. Interfaces that are statically joined are not cleared:

>**enable**
**#show ip igmp groups**
IGMP Connected Group Membership

| Group Address | Interface | Uptime | Expires | Last Reporter |
|---|---|---|---|---|
| 239.192.19.136 | eth 0/1 | 01:38:42 | 00:01:48 | 10.100.13.240 |

# clear ip ospf

Use the **clear ip ospf** command to reset open shortest path first (OSPF) information. Variations of this command include:

**clear ip ospf process**
**clear ip ospf redistribution**

## Syntax Description

| | |
|---|---|
| **process** | Restarts the OSPF process. |
| **redistribution** | Refreshes routes redistributed over OSPF. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example resets the OSPF process:

```
>enable
#clear ip ospf process
```

# clear ip policy-sessions

Use the **clear ip policy-sessions** command to clear policy class sessions. You may clear all the sessions or a specific session. Use the **show ip policy-sessions** command to view a current session listing. The following lists the complete syntax for the **clear ip policy-sessions** commands:

**clear ip policy-sessions**
**clear ip policy-sessions** *<name>* **[ahp | esp | gre | icmp | tcp | udp |** *<protocol>***]** *<source ip>* *<source port>* *<dest ip>* *<dest port>*
**clear ip policy-sessions** *<name>* **[ahp | esp | gre | icmp | tcp | udp |** *<protocol>***]** *<source ip>* *<source port>* *<dest ip>* *<dest port>* **[destination | source]** *<nat ip>* *<nat port>*

## Syntax Description

| | |
|---|---|
| *<name>* | Alphanumeric descriptor for identifying the configured access policy (access policy descriptors are not case-sensitive). |
| **ahp** | Specifies authentication header protocol (AHP). |
| **esp** | Specifies encapsulating security payload protocol (ESP). |
| **gre** | Specifies general routing encapsulation protocol (GRE). |
| **icmp** | Specifies Internet control message protocol (ICMP) protocol. |
| **tcp** | Specifies transmission control protocol (TCP). |
| **udp** | Specifies universal datagram protocol (UDP). |
| *<protocol>* | Specifies a protocol. Valid range is 0 to 255. |
| *<source ip>* | Specifies the source IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| *<source port>* | Specifies the source port (in hex format AHP, ESP, and GRE; decimal for all other protocols). |
| *<dest ip>* | Specifies the destination IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| *<dest port>* | Specifies the destination port (in hex format for AHP, ESP, and GRE; decimal for all other protocols). |
| **[destination \| source]** | For NAT sessions, this specifies whether to select a NAT source or NAT destination session. |
| *<nat ip>* | For NAT sessions, this specifies the NAT IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| *<nat port>* | For NAT sessions, this specifies the NAT port (in hex format for AHP, ESP, and GRE; decimal for all other protocols). |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |

## Functional Notes

The second half of this command, beginning with the source IP address may be copied and pasted from a row in the **show ip policy-sessions** table for easier use.

## Usage Examples

The following example clears the Telnet association (TCP port **23**) for policy class **pclass1** with source IP address **192.22.71.50** and destination **192.22.71.130**:

>**enable**
**#clear ip policy-sessions pclass1 tcp 192.22.71.50 23 192.22.71.130 23**

# clear ip policy-stats

Use the **clear ip policy-stats** command to clear statistical counters for policy classes. Variations of this command include:

**clear ip policy-stats**
**clear ip policy-stats** <*name*>
**clear ip policy-stats** <*name*> **entry** <*number*>

## Syntax Description

| | |
|---|---|
| <*name*> | Optional. Specifies the policy class to clear. If no policy class is specified, statistics are cleared for all policies. |
| **entry** <*number*> | Optional. Clears the statistics of a specific policy class entry. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |

## Usage Examples

The following example clears statistical counters for all policy classes:

>**enable**
**#clear ip policy-stats**

The following example clears statistical counters for the policy class **MatchALL**:

>**enable**
**#clear ip policy-stats MatchALL**

# clear ip prefix-list *<name>*

Use the **clear ip prefix-list** command to clear the IP prefix list hit count shown in the **show ip prefix-list detail** command output. Refer to *show ip prefix-list* on page 287 for more information.

## Syntax Description

| | |
|---|---|
| *<name>* | Clears the count statistics of the specified IP prefix list. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Usage Examples

The following example clears the hit count statistics for prefix list **test**:

```
>enable
#clear ip prefix-list test
```

# clear ip route

Use the **clear ip route** command to remove all learned routes from the IP route table. Static and connected routes are not cleared by this command. Variations of this command include:

**clear ip route ***
**clear ip route** <*ip address*> <*subnet mask*>

## Syntax Description

| | |
|---|---|
| ***** | Deletes all destination routes. |
| <*ip address*> | Specifies the IP address of the destination routes to be deleted. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| <*subnet mask*> | Specifies the subnet mask that corresponds to a range of IP addresses (network) or a specific host. Subnet masks can be expressed in dotted decimal notation (for example, **255.255.255.0**) or as a prefix length (for example, **/24**). |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example removes all learned routes from the route table:

**>enable**
**#clear ip route ****

# clear ip urlfilter statistics

Use the **clear ip urlfilter statistics** command to clear all statistics counters for URL filter requests and responses.

## Syntax Description

No subcommands.

## Default Values

No default necessary for this command.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example clears all counters for URL filter requests and responses:

```
>enable
#clear ip urlfilter statistics
```

# clear lldp counters interface *<interface>*

Use the **clear lldp counters interface** command to reset all local loop demarkation point (LLDP) packet counters to zero on all interfaces.

## Syntax Description

| | |
|---|---|
| *<interface>* | Clears the information for the specified interface. Specify an interface in the format *<interface type [slot/port \| slot/port.sub-interface id \| interface id \| interface id.sub-interface id]>*. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **clear lldp counters interface ?** for a complete list of applicable interfaces. |

## Default Values

There are no default settings for this command.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Usage Examples

The following example resets all LLDP counters:

```
>enable
#clear lldp counters
```

# clear lldp neighbors

Use the **clear lldp neighbors** command to remove all neighbors from this unit's database. As new local loop demarkation point (LLDP) packets are received, the database will contain information about neighbors included in those frames.

## Syntax Description

No subcommands.

## Default Values

There are no default settings for this command.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Functional Notes

This command generates output indicating the names of any neighbors deleted from the database and the name of the interface on which the neighbor was learned.

## Usage Examples

The following example clears LLDP neighbor **Switch_1** from the Ethernet interface 0/7:

```
>enable
#clear lldp neighbors
LLDP: Deleted neighbor "Switch_1" on interface eth 0/7
#
```

# clear mac address-table dynamic

Use the **clear mac address-table dynamic** command to remove dynamic media access control (MAC) addresses from the MAC address table. Variations of this command include:

**clear mac address-table dynamic** <*interface*>
**clear mac address-table dynamic address** <*mac address*>

## Syntax Description

| | |
|---|---|
| <*interface*> | Removes the MAC address of the specified interface. Specify an interface in the format <*interface type [slot/port | slot/port.sub-interface id | interface id | interface id.sub-interface id]*>. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **clear mac address-table dynamic interface ?** for a complete list of applicable interfaces. |
| **address** <*mac address*> | Removes a specific MAC address from the table. MAC addresses should be expressed in following format xx:xx:xx:xx:xx:xx (for example, **00:A0:C8:00:00:01**). |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following example removes the dynamic address **A0:B1:C2:D3:E4:A1** from the MAC address table:

**>enable**
**#clear mac address-table dynamic address A0:B1:C2:D3:E4:A1**

The following example removes all dynamic addresses from the MAC address table:

**>enable**
**#clear mac address-table dynamic**

# clear mac address-table multicast

Use the **clear mac address-table multicast** command to clear all entries in the multicast address resolution lookup (ARL) table or filter the entries based on certain criteria. Variations of this command include:

**clear mac address-table multicast**
**clear mac address-table multicast igmp-snooping**
**clear mac address-table multicast user**
**clear mac address-table multicast vlan** <*vlan id*>
**clear mac address-table multicast vlan** <*vlan id*> **igmp-snooping**
**clear mac address-table multicast vlan** <*vlan id*> **user**

## Syntax Description

| | |
|---|---|
| **igmp-snooping** | Clears entries in the multicast ARL table that were added dynamically (via IGMP snooping). |
| **user** | Clears entries in the multicast ARL table that were added statically (by the user). |
| **vlan** <*vlan id*> | Clears entries in the multicast ARL table based on VLAN. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example removes the entries in the multicast ARL table for VLAN 200:

```
>enable
#clear mac address-table multicast vlan 200
```

# clear port-security

Use the **clear port-security** command to clear the dynamic or sticky secure media access control (MAC) addresses associated with an interface. This can be done on a per-address or per-port basis. Variations of this command include the following:

**clear port-security dynamic address** <*mac address*>
**clear port-security dynamic interface** <*interface*>
**clear port-security sticky address** <*mac address*>
**clear port-security sticky interface** <*interface*>

## Syntax Description

| | |
|---|---|
| **dynamic** | Clears the dynamic MAC addresses. |
| **sticky** | Clears the sticky secure MAC addresses. |
| **address** <*mac address*> | Clears the information for the specified MAC address. MAC addresses should be expressed in following format xx:xx:xx:xx:xx:xx (for example, **00:A0:C8:00:00:01**). |
| **interface** <*interface*> | Clears the information for the specified interface. Specify an interface in the format <*interface type [slot/port* \| *slot/port.sub-interface id* \| *interface id* \| *interface id.sub-interface id]*>. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **clear port-security sticky interface ?** or **clear port-security dynamic interface ?** for a complete list of applicable interfaces. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Usage Examples

The following command clears all dynamic secure MAC addresses associated with the Ethernet interface 0/1:

```
>enable
#clear port-security dynamic interface eth 0/1
```

# clear port-security violation-count *<interface>*

Use the **clear port-security violation-count** command to clear the violation count associated with a particular interface.

## Syntax Description

| | |
|---|---|
| *<interface>* | Clears the information for the specified Ethernet interface. Specify an interface in the format *<interface type [slot/port \| slot/port.sub-interface id \| interface id \| interface id.sub-interface id]>*. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **clear port-security violation-count interface ?** for a complete list of applicable interfaces. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Usage Examples

The following command clears the violation count associated with the Ethernet interface 0/1:

**>enable**
**#clear port-security violation-count eth 0/1**

# clear pppoe <*interface*>

Use the **clear pppoe** command to terminate the current PPPoE client session and cause AOS to attempt to re-establish the session.

## Syntax Description

| | |
|---|---|
| <*interface*> | Specifies the PPP interface ID number to clear. Specify an interface in the format <*interface type [slot/port* \| *slot/port.sub-interface id* \| *interface id* \| *interface id.sub-interface id]*>. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **clear pppoe ?** for a complete list of valid interfaces. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following example ends the current PPPoE client session for ppp 1:

```
>enable
#clear pppoe 1
```

# clear processes cpu max

Use the **clear processes cpu max** command to clear the maximum CPU usage statistic which is displayed in the **show process cpu** command output.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following example resets the CPU maximum usage statistics:

```
>enable
#clear process cpu max
```

# clear processes queue

Use the **clear processes queue** command to clear the contents of the system processing queues.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following example clears the contents of the system processing queues:

```
>enable
#clear process queue
```

# clear qos map

Use the **clear qos map** command to clear the statistics for all defined quality of service (QoS) maps or for maps meeting user-configured specifications. Variations of this command include the following:

**clear qos map**
**clear qos map** <*name*>
**clear qos map** <*name*> <*number*>
**clear qos map interface** <*interface*>

## Syntax Description

| | |
|---|---|
| <*name*> | Optional. Clears the statistics of a defined QoS map. |
| <*number*> | Optional. Clears the statistics for one of the map's specified sequence numbers. |
| **interface** <*interface*> | Optional. Clears QoS map statistics for the specified interface. Specify an interface in the format <*interface type [slot/port* \| *slot/port.sub-interface id* \| *interface id* \| *interface id.sub-interface id]*>. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**.Type **clear qos map interface ?** for a complete list of applicable interfaces. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Usage Examples

The following example clears statistics for all defined QoS maps:

**#clear qos map**

The following example clears statistics for all entries in the **priority** QoS map:

**#clear qos map priority**

The following example clears statistics in entry **10** of the **priority** QoS map:

**#clear qos map priority 10**

The following example clears QoS statistics for a specified interface:

**#clear qos map interface frame-relay 1**


*The **clear counters** command clears ALL interface statistics (including QoS map interface statistics).*

# clear relay

Use the **clear relay** command to reset the door contact relay.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example resets the door contact relay:

```
>enable
#clear relay
```

# clear route-map counters

Use the **clear route-map counters** command to reset route map hit counters. Variations of this command include:

**clear route-map counters**
**clear route-map counters** <*name*>

## Syntax Description

| | |
|---|---|
| <*name*> | Optional. Clears the counters for the specified route map. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example clears all route map counters:

**>enable**
**#clear route-map counters**

# clear sip location *<username>*

Use the **clear sip location** command to clear session initiation protocol (SIP) location database statistics.

## Syntax Description

| | |
|---|---|
| *<username>* | Clears the statistics for the specified user name. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example deletes all dynamic location entries:

```
>enable
#clear sip location **
```

# clear sip trunk-registration

Use the **clear sip trunk-registration** command to clear local Session Initiation Protocol (SIP) registration information for one or more trunks. Variations of this command include:

**clear sip trunk-registration**
**clear sip trunk-registration** <*Txx*>
**clear sip trunk-registration** <*Txx*> <*identity*>

## Syntax Description

| | |
|---|---|
| <*Txx*> | Optional. Specifies the trunk to clear using its two-digit identifier. For example: T01. |
| <*identity*> | Optional. Specifies the identity of the trunk registration to clear. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example clears SIP registration information for trunk 01:

**>enable**
**#clear sip trunk-registration T01**

# clear sip user-registration

Use the **clear sip user-registration** command to clear local session initiation protocol (SIP) server registration information.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example clears all SIP server registration information:

```
>enable
#clear sip user-registration
```

# clear spanning-tree counters

The **clear spanning-tree counters** command clears the following counts: BPDU transmit, BPDU receive, and number of transitions to forwarding state. Variations of this command include:

**clear spanning-tree counters**
**clear spanning-tree counters interface** *<interface>*

## Syntax Description

| | |
|---|---|
| **interface** *<interface>* | Optional. Specifies a single interface. Specify an interface in the format *<interface type [slot/port \| slot/port.sub-interface id \| interface id \| interface id.sub-interface id]>*. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **clear spanning-tree counters ?** for a complete list of interfaces. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Usage Examples

The following example clears the spanning tree counters for Ethernet 0/10:

>**enable**
**#clear spanning-tree counters interface eth 0/10**

# clear spanning-tree detected-protocols

Use the **clear spanning-tree detected-protocols** command to restart the protocol migration process. Variations of this command include:

**clear spanning-tree detected-protocols**
**clear spanning-tree detected-protocols interface** <*interface*>

## Syntax Description

| | |
|---|---|
| **interface** <*interface*> | Optional. Specifies a valid interface to clear. Specify an interface in the format <*interface type [slot/port | slot/port.sub-interface id | interface id | interface id.sub-interface id]*>. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **clear spanning-tree detected-protocols interface ?** for a complete list of applicable interfaces. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

The switch has the ability to operate using the rapid spanning-tree protocol or the legacy 802.1D version of spanning-tree. When a BPDU (bridge protocol data unit) of the legacy version is detected on an interface, the switch automatically regresses to using the 802.1D spanning-tree protocol for that interface. Issue the **clear spanning-tree detected-protocols** command to return to rapid spanning-tree operation.

## Usage Examples

The following example re-initiates the protocol migration process on Ethernet interface 0/3:

>**enable**
**#clear spanning-tree detected-protocols interface ethernet 0/3**

The following example re-initiates the protocol migration process on all interfaces:

>**enable**
**#clear spanning-tree detected-protocols**

# clear tacacs+ statistics

Use the **clear tacacs+ statistics** command to delete all terminal access controller access control system (TACACS+) protocol statistics.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example clears all TACACS+ protocol statistics:

```
>enable
#clear tacacs+ statistics
```

# clear user

Use the **clear user** command to detach a user from a given line. Variations of this command include:

**clear user console** <*number*>
**clear user ssh** <*number*>
**clear user telnet** <*number*>

## Syntax Description

| | |
|---|---|
| **console** <*number*> | Detaches a specific console user. Valid range is 0 to 1. |
| **ssh** <*number*> | Detaches a specific secure shell (SSH) user. Valid range is 0 to 4. |
| **telnet** <*number*> | Detaches a specific Telnet user. Valid range is 0 to 5. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example detaches the **console 1** user:

>**enable**
**#clear user console 1**

# clock auto-correct-dst

The **clock auto-correct-dst** command allows the automatic one-hour correction for Daylight Saving Time (DST). Use the **clock no-auto-correct-dst** command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default this command is enabled.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Usage Examples

The following example allows for automatic DST correction:

```
>enable
#clock auto-correct-dst
```

# clock no-auto-correct-dst

The **clock no-auto-correct-dst** command allows you to override the automatic one-hour correction for Daylight Saving Time (DST).

## Syntax Description

No subcommands.

## Default Values

No default value is necessary for this command.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Functional Notes

Many time zones include an automatic one-hour correction for daylight saving time at the appropriate time. You may override it at your location using this command.

## Usage Examples

The following example overrides the one-hour offset for DST:

```
>enable
#clock no-auto-correct-dst
```

# clock set *<time> <day> <month> <year>*

Use the **clock set** command to configure the system software clock. For the command to be valid, all fields must be entered. Refer to the *Usage Examples* below for an example.

## Syntax Description

| | |
|---|---|
| *<time>* | Sets the time (in 24-hour format) of the system software clock in the format HH:MM:SS (hours:minutes:seconds). |
| *<day>* | Sets the current day of the month. Valid range is 1 to 31. |
| *<month>* | Sets the current month. Valid range is January to December. You need only enter enough characters to make the entry unique. This entry is not case-sensitive. |
| *<year>* | Sets the current year. Valid range is 2000 to 2100. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example sets the system software clock for 3:42 pm, August 22 2004:

>**enable**
**#clock set 15:42:00 22 Au 2004**

## clock timezone <*value*>

The **clock timezone** command sets the unit's internal clock to the timezone of your choice. This setting is based on the difference in time (in hours) between Greenwich Mean Time (GMT) or Central Standard Time (CST) and the timezone for which you are setting up the unit. Use the **no** form of this command to disable this feature.

### Syntax Description

| | |
|---|---|
| <*value*> | Clock timezone values are specified in the *Functional Notes* section for this command. |

### Default Values

No default value is necessary for this command.

### Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |
| Release 11.1 | Command was expanded to include clock timezone 0. |

## Functional Notes

The following list shows sample cities and their timezone codes.

clock timezone +1-Amsterdam
clock timezone +1-Belgrade
clock timezone +1-Brussels
clock timezone +1-Sarajevo
clock timezone +1-West-Africa
clock timezone +10-Brisbane
clock timezone +10-Canberra
clock timezone +10-Guam
clock timezone +10-Hobart
clock timezone +10-Vladivostok
clock timezone +11
clock timezone +12-Auckland
clock timezone +12-Fiji
clock timezone +13
clock timezone +2-Athens
clock timezone +2-Bucharest
clock timezone +2-Cairo
clock timezone +2-Harare
clock timezone +2-Helsinki
clock timezone +2-Jerusalem
clock timezone +3-Baghdad
clock timezone +3-Kuwait
clock timezone +3-Moscow
clock timezone +3-Nairobi
clock timezone +3:30
clock timezone +4-Abu-Dhabi
clock timezone +4-Baku
clock timezone +4:30
clock timezone +5-Ekaterinburg
clock timezone +5-Islamabad
clock timezone +5:30
clock timezone +5:45
clock timezone +6-Almaty
clock timezone +6-Astana
clock timezone +6-Sri-Jay
clock timezone +6:30
clock timezone +7-Bangkok
clock timezone +7-Kranoyarsk
clock timezone +8-Bejing
clock timezone +8-Irkutsk
clock timezone +8-Kuala-Lumpur
clock timezone +8-Perth
clock timezone +8-Taipei
clock timezone +9-Osaka
clock timezone +9-Seoul
clock timezone +9-Yakutsk
clock timezone +9:30-Adelaide
clock timezone +9:30-Darwin
clock timezone -1-Azores
clock timezone -1-Cape-Verde
clock timezone -10
clock timezone -11
clock timezone -12
clock timezone -2
clock timezone -3-Brasilia
clock timezone -3-Buenos-Aires
clock timezone -3-Greenland
clock timezone -3:30
clock timezone -4-Atlantic-Time
clock timezone -4-Caracus
clock timezone -4-Santiago
clock timezone -5
clock timezone -5-Bogota
clock timezone -5-Eastern-Time
clock timezone -6-Central-America
clock timezone -6-Central-Time
clock timezone -6-Mexico-City
clock timezone -6-Saskatchewan
clock timezone -7-Arizona
clock timezone -7-Mountain-Time
clock timezone -8
clock timezone -9
clock timezone 0-(Universal Coordinated Time (UTC)
clock timezone GMT-Casablanca
clock timezone GMT-Dublin

## Usage Examples

The following example sets the timezone for Santiago, Chile.

>**enable**
**#clock timezone -4-Santiago**

# configure

Use the **configure** command to enter the Global Configuration mode or to configure the system from memory. Refer to *Global Configuration Mode Command Set* on page 417 for more information. Variations of this command include:

**configure memory**
**configure network**
**configure overwrite-network**
**configure terminal**

## Syntax Description

| | |
|---|---|
| **memory** | Configures the active system with the commands located in the default configuration file stored in NVRAM. |
| **network** | Configures the system from a TFTP network host. |
| **overwrite-network** | Overwrites NVRAM memory from a TFTP network host. |
| **terminal** | Enters the Global Configuration mode. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example enters the Global Configuration mode from the Enable mode:

```
>enable
#configure terminal
(config)#
```

# **copy** *<source> <destination>*

Use the **copy** command to copy any file from a specified source to a specified destination.

## Syntax Description

| | |
|---|---|
| *<source>* | Specifies the current location of the file to copy.<br>Valid sources include: **running-config** (current running configuration file), **startup-config** (configuration file located in NVRAM), or a filename (located in FLASH memory). |
| *<destination>* | Specifies the destination of the copied file.<br>Valid destinations include: **running-config** (current running configuration file), **startup-config** (configuration file located in NVRAM), or a filename (located in FLASH memory). |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example creates a copy of the file **myfile.biz** (located in FLASH memory) and names it **newfile.biz**:

>**enable**
**#copy myfile.biz newfile.biz**

The following example creates a backup copy of the startup configuration file (and places in FLASH memory):

>**enable**
**#copy startup-config backup.bak**

The following example copies the current running-configuration file to the startup configuration file located in NVRAM:

>**enable**
**#copy running-config startup-config**

# copy console *<filename>*

Use the **copy console** command to copy the console's input to a text file. To end copying to the text file, type **<Ctrl+D>**. The file will be saved in the AOS root directory.

## Syntax Description

| | |
|---|---|
| *<filename>* | Specifies destination file for console input. |

## Default Values

No default is necessary for this command.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Functional Notes

The copy console command works much like a line editor. Prior to pressing **<Enter>**, changes can be made to the text on the line. Changes can be made using **<Delete>** and **<Backspace>** keys. The text can be traversed using the arrow keys, **<Ctrl+A>** (to go to the beginning of a line), and **<Ctrl+E>** (to go to the end of a line). To end copying to the text file, type **<Ctrl+D>**. The file will be saved in the AOS root directory. Use the **dir** command to see a list of files in the root directory.

## Usage Examples

The following example copies the console input into the file **config** (located in the AOS root directory):

>**enable**
**#copy console config**

# copy flash *<destination>*

Use the **copy flash** command to copy a file located in flash memory to a specified destination.

## Syntax Description

| | |
|---|---|
| *<destination>* | Specifies the destination of the copied file. Valid destinations include **tftp** and **xmodem**. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example copies the contents of the unit's flash memory to a TFTP server:

```
>enable
#copy flash tftp
```

# **copy** <*filename*> **interface** <*interface*>

Use the **copy interface** command to copy a file to a specified interface.

## Syntax Description

| | |
|---|---|
| <*filename*> | Specifies file name of source file to copy. |
| <*interface*> | Specifies interface to be upgraded. Specify an interface in the format <*interface type [slot/port* \| *slot/port.sub-interface id* \| *interface id* \| *interface id.sub-interface id]*>. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **copy** <*filename*> **interface ?** for a complete list of valid interfaces. |

## Default Values

No default is necessary for this command.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Usage Examples

The following example upgrades the ADSL interface with the firmware file **configfile**:

```
>enable
#copy configfile interface adsl 0/1
```

# copy tftp *<destination>*

Use the **copy tftp** command to copy a file located on a network Trivial File Transfer Protocol (TFTP) server to a specified destination.

## Syntax Description

| | |
|---|---|
| *<destination>* | Specifies the destination of the file copied from the TFTP server.<br>Valid destinations include: **flash** (Flash memory), **startup-config** (the configuration file stored in NVRAM), or **running-config** (the current running configuration file).<br>After entering **copy tftp** and specifying a destination, AOS prompts for the following information: |
| *Address of remote host:* | IP address of the TFTP server. |
| *Source filename:* | Name of the file to copy from the TFTP server. |
| *Destination filename:* | Specifies the filename to use when storing the copied file to Flash memory. (Valid only for the **copy tftp flash** command.) |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example copies **myfile.biz** from the TFTP server (10.200.2.4) to flash memory and labels it **newfile.biz**:

>**enable**
**#copy tftp flash**
Address of remote host?**10.200.2.4**
Source filename **myfile.biz**
Destination filename **newfile.biz**
Initiating TFTP transfer...
Received 45647 bytes.
Transfer Complete!
#

# copy xmodem <*destination*>

Use the **copy xmodem** command to copy a file (using the XMODEM protocol) to a specified destination. XMODEM capability is provided in terminal emulation software such as HyperTerminal™.

## Syntax Description

| | |
|---|---|
| *<destination>* | Specifies the destination of the copied file.<br>Valid destinations include: **flash** (Flash memory), **startup-config** (the configuration file stored in NVRAM), or **running-config** (the current running configuration file).<br>After entering **copy xmodem** and specifying a destination, AOS prompts for the following information: |
| *Destination filename:* | Specifies the filename to use when storing the copied file to Flash memory. (Valid only for the **copy flash** command.) |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example copies a .biz file to flash memory and labels it **newfile.biz**:

**>enable**
**#copy xmodem flash**
Destination filename **newfile.biz**
Begin the Xmodem transfer now...
Press CTRL+X twice to cancel
CCCCCC

AOS is now ready to accept the file on the **CONSOLE** port (using the XMODEM protocol). The next step in the process may differ depending on the type of terminal emulation software you are using. For HyperTerminal, you will now select **Transfer > Send File** and browse to the file you wish to copy. Once the transfer is complete, information similar to the following is displayed:

Received 231424 bytes.
Transfer complete.

# debug aaa

Use the **debug aaa** command to activate debug messages associated with authentication from the AAA subsystem. Debug messages are displayed (real time) on the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages.

*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

No subcommands.

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

The **debug aaa** events include connection notices, login attempts, and session tracking.

## Usage Examples

The following is sample output for this command:

```
>enable
#debug aaa
AAA: New Session on portal 'TELNET 0 (172.22.12.60:4867)'.
AAA: No list mapped to 'TELNET 0'. Using 'default'.
AAA: Attempting authentication (username/password).
AAA: RADIUS authentication failed.
AAA: Authentication failed.
AAA: Closing Session on portal 'TELNET 0 (172.22.12.60:4867)'.
```

# debug access-list *<name>*

Use the **debug access-list** command to activate debug messages (for a specified list) associated with access control list operation. Debug messages are displayed (real time) on the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages.

*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies a configured access control list. |

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Functional Notes

The **debug access-list** command provides debug messages to aid in troubleshooting access control list issues.

## Usage Examples

The following example activates debug messages for the access control list labeled **MatchAll**:

```
>enable
#debug access-list MatchAll
```

# debug arp

Use the **debug arp** command to activate debug messages associated with IP Address Resolution Protocol (ARP) transactions. Debug messages are displayed (real time) on the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages.


*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

No subcommands.

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example activates debug messages associated with ARP transactions:

```
>enable
#debug arp
```

# debug atm events

Use the **debug atm events** command to display events on all ATM ports and all virtual circuits. Debug messages are displayed (real time) to the terminal (or Telnet) screen. Use the **no** form of this command to disable debug messages.


*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

No subcommands.

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Usage Examples

The following example activates ATM event messages:

```
>enable
#debug atm events
```

# debug atm oam

Use the **debug atm oam** command to display Operation, Administration, and Maintenance (OAM) packets for an ATM virtual circuit descriptor (VCD). Debug messages are displayed (real time) to the terminal (or Telnet) screen. Use the **no** form of this command to disable debug messages. Variations of this command include the following:

**debug atm oam**
**debug atm oam** <*vcd*>
**debug atm oam** <*vcd*> **loopback end-to-end**
**debug atm oam** <*vcd*> **loopback end-to-end** <*LLID*>
**debug atm oam** <*vcd*> **loopback segment**
**debug atm oam** <*vcd*> **loopback segment** <*LLID*>


*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

| | |
|---|---|
| <*vcd*> | Optional. Shows OAM packets for a specific VCD. |
| **loopback** | Optional. Configures an OAM loopback. |
| **end-to-end** | Optional. Configures an end-to-end OAM loopback. |
| **segment** | Optional. Configures a segment loopback. |
| **<LLID>** | Optional. Specifies 16-byte OAM loopback location ID (LLID). |

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Usage Examples

The following example activates ATM OAM debug messages for VCD 1:

**>enable**
**#debug atm oam 1**

# debug atm packet

Use the **debug atm packet** command to activate debug messages associated with packets on ATM ports and virtual circuits. Debug messages are displayed (real time) to the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages. Variations of this command include the following:

**debug atm packet**
**debug atm packet interface atm** <*port id*>
**debug atm packet interface atm** <*port id*> **vcd** <*number*>
**debug atm packet vc** <*VPI/VCI*>

*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

| | |
|---|---|
| **interface atm** <*port id*> | Shows packets on a specific ATM port and on all virtual circuits. |
| **vc** <*VPI/VCI*> | Shows packets on a specific virtual circuit identified by the virtual path identifier and virtual channel identifier (VPI/VCI). |
| **vcd** <*number*> | Shows packets on specific virtual circuit descriptors (VCD). |

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Usage Examples

The following example activates debug ATM packet debug messages on ATM port 1:

**>enable**
**#debug atm packet interface atm 1**

# debug auto-config

Use the **debug auto-config** command to activate debug messages associated auto-config events. Debug messages are displayed (real time) on the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages.


*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

No subcommands.

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The example activates debug messages associated with auto-config events:

```
>enable
#debug auto-config
```

# debug bridge

Use the **debug bridge** command to display messages associated with bridge events. Debug messages are displayed (real time) to the terminal (or Telnet) screen. Use the **no** form of this command to disable debug messages.


*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

No subcommands.

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Usage Examples

The following example activates bridge debug messages:

```
>enable
#debug bridge
```

# debug chat-interfaces *<chat interface>*

Use the **debug chat-interfaces** command to activate debug messages associated with chat AT command driven interfaces. Debug messages are displayed (real time) on the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages.


*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

| | |
|---|---|
| *<chat interface>* | Specifies the chat interface to debug in slot/port format. |

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example activates debug messages for the chat interface 0/1:

**>enable**
**#debug chat-interfaces 0/1**

# debug crypto

Use the **debug crypto** command to activate debug messages associated with IKE and IPSec functions. Debug messages are displayed (real time) to the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages.Variations of this command include:

**debug crypto ike**
**debug crypto ike client authentication**
**debug crypto ike client configuration**
**debug crypto ike negotiation**
**debug crypto ipsec**
**debug crypto pki**

*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

| | |
|---|---|
| **ike** | Displays all IKE debug messages. |
| **ike client authentication** | Displays IKE client authentication messages as they occur. |
| **ike client configuration** | Displays mode-config exchanges as they take place over the IKE SA. It is enabled independently from the **ike negotiation** debug described previously. |
| **ike negotiation** | Displays only IKE key management debug messages (e.g., handshaking). |
| **ipsec** | Displays all IPSec debug messages. |
| **pki** | Displays all public key infrastructure (PKI) debug messages. |

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 4.1 | Command was introduced. |
| Release 6.1 | Debug pki command introduced. |

## Usage Examples

The following example activates the IPSec debug messages:

**>enable**
**#debug crypto ipsec**

# debug data-call

Use the **debug data-call** command to activate debug messages associated with data call errors and events. Debug messages are displayed (real time) on the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages.


*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

No subcommands.

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example activates debug messages associated with data call errors and events:

```
>enable
#debug data-call
```

# debug demand-routing

Use the **debug demand-routing** command to activate debug messages associated with demand routing errors and events. Debug messages are displayed (real time) to the terminal (or Telnet) screen. Use the **no** form of this command to disable debug messages.


*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

No subcommands.

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example activates demand routing error and event messages:

```
>enable
#debug demand-routing
```

# debug dial-backup

Use the **debug dial-backup** command to activate debug messages associated with dial-backup operation. Debug messages are displayed (real time) to the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages.

*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

No subcommands.

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 2.1 | Additional debug messages were implemented for dial-backup operation to ADTRAN's IQ and Express Series products. |

## Functional Notes

The **debug dial-backup** command activates debug messages to aid in the troubleshooting of dial-backup links.

## Usage Examples

The following example activates debug messages for dial-backup operation:

```
>enable
#debug dial-backup
```

# debug dialup-interfaces

Use the **debug dialup-interfaces** command to generate debug messages used to aid in troubleshooting problems with all dialup interfaces such as the modem or the BRI cards. Debug messages are displayed (real time) on the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages.


*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

No subcommands.

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |

## Functional Notes

When enabled, these messages provide status information on incoming calls, dialing and answering progress, etc. These messages also give information on why certain calls are dropped or rejected. It is beneficial to use this command when troubleshooting dial backup (in addition to the **debug dial-backup** command).

## Usage Examples

The following example activates the debug messages for dialup interfaces:

```
>enable
#debug dialup-interfaces
```

# debug dynamic-dns

Use the **debug dynamic-dns** command to display debug messages associated with dynamic domain naming system (DNS). Debug messages are displayed (real time) to the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages. Variations of this command include:

**debug dynamic-dns**
**debug dynamic-dns verbose**

*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

| | |
|---|---|
| **verbose** | Optional. Enables detailed debug messages. |

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example activates dynamic DNS debug messages:

**>enable**
**#debug dynamic-dns verbose**

# debug firewall

Use the **debug firewall** command to activate debug messages associated with the AOS firewall operation. Debug messages are displayed (real time) to the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages.


*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

No subcommands.

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |

## Functional Notes

The **debug firewall** command activates debug messages to provide real-time information about the AOS stateful inspection firewall operation.

## Usage Examples

The following example activates the debug messages for the AOS stateful inspection firewall:

```
>enable
#debug firewall
```

# debug firewall alg sip

Use the **debug firewall alg sip** command to activate debug messages associated with Session Initiation Protocol (SIP) information with AOS firewall operation. Debug messages are displayed (real time) on the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages. Variations of this command include:

**debug firewall alg sip**
**debug firewall alg sip verbose**

*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

| | |
|---|---|
| **verbose** | Optional. Enables detailed debug messages. |

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example activates debug messages associated with SIP information with AOS firewall operation:

**>enable**
**#debug firewall alg sip**

# debug frame-relay

Use the **debug frame-relay** command to activate debug messages associated with the Frame Relay operation. Debug messages are displayed (real time) to the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages. Variations of this command include:

**debug frame-relay events**
**debug frame-relay llc2**
**debug frame-relay lmi**


*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

| | |
|---|---|
| **events** | Activates debug messages for generic Frame Relay events (such as Frame Relay interface state). |
| **llc2** | Activates debug messages for the logical link control layer. |
| **lmi** | Activates debug messages for the local management interface (such as DLCI status signaling state, etc.). |

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

The **debug frame-relay** command activates debug messages to aid in the troubleshooting of Frame Relay links.

## Usage Examples

The following example activates all possible debug messages associated with Frame Relay operation:

>**enable**
**#debug frame-relay events**
**#debug frame-relay llc2**
**#debug frame-relay lmi**

# debug frame-relay multilink

Use the **debug frame-relay multilink** command to activate debug messages associated with Frame Relay multilink operation. Debug messages are displayed (real time) to the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages. Variations of this command include:

**debug frame-relay multilink**
**debug frame-relay multilink** <*interface*>
**debug frame-relay multilink states**


*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

| | |
|---|---|
| <*interface*> | Optional. Specifies an interface in the format <*interface type [slot/port | slot/port.sub-interface id | interface id | interface id.sub-interface id]*>. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp** 1; and for an ATM sub-interface use **atm 1.1**. Type **debug frame-relay multilink ?** for a complete list of applicable interfaces. |
| **states** | Optional. Activates the debug messages for Link Integrity Protocol (LIP). |

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Usage Examples

The following example activates debug messages associated with multilink operation for all Frame Relay interfaces:

**>enable**
**#debug frame-relay multilink**

# debug gvrp bpdus

Use the **debug gvrp bpdus** command to see debug messages showing all GARP VLAN Registration Protocol (GVRP) configuration messages sent and received on the switch. Debug messages are displayed (real time) on the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages.


*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

No subcommands.

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Functional Notes

With GVRP enabled on many ports, this command can produce a lot of output. To see these messages just for individual interfaces, refer to the command *debug gvrp interface <interface>* on page 121.

## Usage Examples

The following example displays debug messages showing GVRP configuration messages sent and received on Ethernet interface 0/24:

```
>enable
#debug gvrp bpdus
2000.07.31 23:15:51 GVRP BPDUS.eth 0/24: TX = (Len:2 LeaveAll) (Len:4 JoinIn Vlan:1) (End) ... SENT
2000.07.31 23:15:52 GVRP BPDUS.eth 0/24: RX = (Len:4 Empty Vlan:2) (Len:4 JoinIn Vlan:20) (end)
2000.07.31 23:15:52 GVRP BPDUS.eth 0/24: TX = (Len:4 JoinIn Vlan:1) (End) ... SENT
2000.07.31 23:15:52 GVRP BPDUS.eth 0/24: RX = (Len:4 JoinIn Vlan:20) (end)
2000.07.31 23:16:00 GVRP BPDUS.eth 0/24: RX = (Len:2 LeaveAll) (end)
#
```

# debug gvrp interface *<interface>*

Use the **debug gvrp interface** command to see GARP VLAN Registration Protocol (GVRP) debug messages related to a particular interface. Debug messages are displayed (real time) on the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages. Variations of this command include:

**debug gvrp interface** *<interface>* **bpdus**
**debug gvrp interface** *<interface>* **vlans**

*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

| | |
|---|---|
| *<interface>* | Activates debug messages for the specified interface. Specify an interface in the format *<interface type [slot/port \| slot/port.sub-interface id \| interface id \| interface id.sub-interface id]>*. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **debug gvrp interface ?** for a complete list of applicable interfaces. |
| **bpdus** | Displays debug messages showing all GVRP configuration messages sent and received on the interface. |
| **vlans** | Displays debug messages showing all GVRP-related VLAN changes occurring on the interface. |

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Usage Examples

The following example displays debug messages showing GVRP configuration messages sent and received on Ethernet interface 0/24:

```
>enable
#debug gvrp interface ethernet 0/24 bpdus
2000.07.31 23:15:51 GVRP BPDUS.eth 0/24: TX = (Len:2 LeaveAll) (Len:4 JoinIn Vlan:1) (End) ... SENT
2000.07.31 23:15:52 GVRP BPDUS.eth 0/24: RX = (Len:4 Empty Vlan:2) (Len:4 JoinIn Vlan:20) (end)
2000.07.31 23:15:52 GVRP BPDUS.eth 0/24: TX = (Len:4 JoinIn Vlan:1) (End) ... SENT
2000.07.31 23:15:52 GVRP BPDUS.eth 0/24: RX = (Len:4 JoinIn Vlan:20) (end)
2000.07.31 23:16:00 GVRP BPDUS.eth 0/24: RX = (Len:2 LeaveAll) (end)
```

# debug gvrp vlans

Use the **debug gvrp vlans** command to see debug messages showing all GARP VLAN Registration Protocol (GVRP) VLAN changes. Debug messages are displayed (real time) on the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages. Variations of this command include:

**debug gvrp vlans**
**debug gvrp vlans** <*vlan id*>


*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

| | |
|---|---|
| <*vlan id*> | Optional. Displays debug messages showing all GVRP-related VLAN changes for this VLAN only. Range is 1 to 4094. |

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Functional Notes

With GVRP enabled on many ports, this command can produce a lot of output. To see these messages just for individual interfaces, refer to the command *debug gvrp interface <interface>* on page 121.

## Usage Examples

The following example displays debug messages showing GVRP-related VLAN changes for VLAN 1:

```
>enable
#debug gvrp vlans 1
#
2000.07.31 22:05:42 GVRP VLANS: Creating dynamic VLAN 20
2000.07.31 22:05:42 GVRP VLANS.eth 0/24: Dynamically adding port to VLAN 20
#
2000.07.31 22:05:56 INTERFACE_STATUS.eth 0/24 changed state to down
2000.07.31 22:06:08 GVRP VLANS.eth 0/24: Dynamically removing port from VLAN 20
2000.07.31 22:06:08 GVRP VLANS: Last port removed from VLAN 20, destroying VLAN
```

# debug interface <*interface*>

Use the **debug interface** command to activate debug messages associated with the specified interface. Debug messages are displayed (real time) to the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages.

*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

| | |
|---|---|
| <*interface*> | Activates debug messages for the specified interface. Specify an interface in the format <*interface type [slot/port | slot/port.sub-interface id | interface id | interface id.sub-interface id]*>. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **debug interface ?** for a complete list of applicable interfaces. |

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |
| Release 6.1 | Command was expanded to include T1 and FXS interfaces. |
| Release 7.1 | Command was expanded to include FXO interface. |
| Release 9.1 | Command was expanded to include tunnel interface. |

## Functional Notes

The **debug interface** command activates debug messages to aid in the troubleshooting of physical interfaces.

## Usage Examples

The following example activates all possible debug messages associated with the Ethernet port:

```
>enable
#debug interface ethernet
```

# debug interface adsl events

Use the **debug interface adsl events** command to activate debug messages associated with ADSL events. Debug messages are displayed (real time) to the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages.


*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

No subcommands.

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Usage Examples

The following example activates debug messages for ADSL events:

```
>enable
#debug interface adsl events
```

# debug ip bgp

Use the **debug ip bgp** command to activate debug messages associated with IP Border Gateway Protocol (BGP). Debug messages are displayed (real time) on the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages. Variations of this command include:

**debug ip bgp**
**debug ip bgp events**
**debug ip bgp in**
**debug ip bgp out**
**debug ip bgp keepalives**
**debug ip bgp updates**
**debug ip bgp updates quiet**


*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

| | |
|---|---|
| **events** | Optional. Displays significant BGP events such as a neighbor state change. |
| **in/out** | Optional. Displays the same information as **debug ip bgp**, but limits messages to the specified direction (in or out). |
| **keepalives** | Optional. Displays BGP keepalive packets. |
| **updates** | Optional. Displays detailed information on BGP updates for all neighbors. |
| **updates quiet** | Optional. Displays summary information about BGP neighbor updates. (Note: **updates quiet** displays a one-line summary of what **update** displays in 104 lines.) |

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Functional Notes

If no arguments are given, the **debug ip bgp** command displays general BGP events such as sent/received message summaries, route processing actions, and results. Keepalive packets are not debugged with this command.

## Usage Examples

The following example enables debug messages on general outbound BGP messages and events:

>**enable**
**#debug ip bgp out**
#07:42:39: BGP OUT 10.15.240.1[2]: Transmitting msg, type=UPDATE (2), len=142

# debug ip dhcp-client

Use the **debug ip dhcp-client** command to activate debug messages associated with Dynamic Host Configuration Protocol (DHCP) client operation in AOS. Debug messages are displayed (real time) to the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages.

*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

No subcommands.

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |

## Functional Notes

The **debug ip dhcp-client** command activates debug messages to provide information on DHCP client activity in AOS. The AOS DHCP client capability allows interfaces to dynamically obtain an IP address from a network DHCP server.

## Usage Examples

The following example activates debug messages associated with DHCP client activity:

```
>enable
#debug ip dhcp-client
```

# debug ip dhcp-server

Use the **debug ip dhcp-server** command to activate debug messages associated with Dynamic Host Configuration Protocol (DHCP) server operation in AOS. Debug messages are displayed (real time) to the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages.


*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

No subcommands.

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |

## Functional Notes

The **debug ip dhcp-server** command activates debug messages to provide information on DHCP server activity in AOS. The AOS DHCP server capability allows AOS to dynamically assign IP addresses to hosts on the network.

## Usage Examples

The following example activates debug messages associated with DHCP server activity:

```
>enable
#debug ip dhcp-server
```

# debug ip dns-client

Use the **debug ip dns-client** command to activate debug messages associated with domain naming system (DNS) client operation in AOS. Debug messages are displayed (real time) to the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages.


*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

No subcommands.

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Functional Notes

The **debug ip dns-client** command activates debug messages to provide information on DNS client activity in AOS. The IP DNS capability allows for DNS-based host translation (name-to-address).

## Usage Examples

The following example activates debug messages associated with DNS client activity:

```
>enable
#debug ip dns-client
```

# debug ip dns-proxy

Use the **debug ip dns-proxy** command to activate debug messages associated with domain naming system (DNS) proxy operation in AOS. Debug messages are displayed (real time) to the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages.


*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

No subcommands.

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Functional Notes

The **debug ip dns-proxy** command activates debug messages to provide information on DNS proxy activity in AOS. The IP DNS capability allows for DNS-based host translation (name-to-address).

## Usage Examples

The following example activates debug messages associated with DNS proxy activity:

```
>enable
#debug ip dns-proxy
```

# debug ip ftp-server

Use the **debug ip ftp-server** command to activate debug messages associated with File Transfer Protocol (FTP) server events in the AOS device. Debug messages are displayed (real time) to the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages.


*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

No subcommands.

## Default Values

By default, all debug messages in the AOS are disabled.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Functional Notes

The **debug ip ftp-server** command activates debug messages to provide information on FTP server activity in the AOS. The FTP server capability allows for fast file management and transport for local or remote devices.

## Usage Examples

The following example activates debug messages associated with FTP server activity:

```
>enable
#debug ip ftp-server
```

# debug ip http

Use the **debug ip http** command to activate debug messages associated with HyperText Transfer Protocol (HTTP) operation in AOS. Debug messages are displayed (real time) to the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages. Variations of this command include:

**debug ip http**
**debug ip http verbose**


*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

| | |
|---|---|
| **verbose** | Optional. Activates detailed debug messages for HTTP operation. |

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Usage Examples

The following example activates debug messages associated with HTTP activity:

**>enable**
**#debug ip http**

# debug ip icmp

Use the **debug ip icmp** command to show all Internet Control Message Protocol (ICMP) messages as they come into the router or are originated by the router. If an optional keyword (**send** or **recv**) is not used, all results are displayed. Debug messages are displayed (real time) on the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages. Variations of this command include:

**debug ip icmp**
**debug ip icmp send**
**debug ip icmp recv**


*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

| | |
|---|---|
| **send** | Optional. Displays only ICMP messages sent by the router. |
| **recv** | Optional. Displays only ICMP messages received by the router. |

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Usage Examples

The following example activates the **debug ip icmp** send and receive messages for AOS:

```
>enable
#debug ip icmp
ICMP SEND: From (0.0.0.0) to (172.22.14.229) Type=8 Code=0 Length=72 Details:echo request
ICMP RECV: From (172.22.14.229) to (10.100.23.19) Type=0 Code=0 Length=72 Details:echo reply
ICMP SEND: From (0.0.0.0) to (172.22.14.229) Type=8 Code=0 Length=72 Details:echo request
ICMP RECV: From (172.22.14.229) to (10.100.23.19) Type=0 Code=0 Length=72 Details:echo reply
ICMP RECV: From (172.22.255.200) to (10.100.23.19) Type=11 Code=0 Length=36 Details:TTL equals 0
    during transit
ICMP RECV: From (172.22.14.229) to (10.100.23.19) Type=3 Code=3 Length=36 Details:port
    unreachable
ICMP RECV: From (172.22.14.229) to (10.100.23.19) Type=3 Code=3 Length=36 Details:port
    unreachable
```

# debug ip igmp

Use the **debug ip igmp** command to enable debug messages for Internet Group Management Protocol (IGMP) transactions (including helper activity). Debug messages are displayed (real time) on the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages. Variations of this command include:

**debug ip igmp**
**debug ip igmp** <*ip address*>

*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

| | |
|---|---|
| <*ip address*> | Optional. Specifies the IP address of a multicast group. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |

## Usage Examples

The following example enables IGMP debug messages for the specified multicast group:

**>enable**
**#debug ip igmp 224.1.1.1**

# debug ip igmp snooping

Use the **debug ip igmp snooping** command to enable debug messages for Internet Group Management Protocol (IGMP) snooping errors and events. Debug messages are displayed (real time) on the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages. Variations of this command include:

**debug ip igmp snooping**
**debug ip igmp snooping verbose**

*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

| | |
|---|---|
| **verbose** | Optional. Enables detailed debug messages. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example enables IGMP snooping debug messages:

**>enable**
**#debug ip igmp snooping**

# debug ip mrouting

Use the **debug ip mrouting** command to activate debug messages associated with multicast table routing events. Debug messages are displayed (real time) on the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages.


*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

No subcommands.

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following sample activates **ip mrouting** debug messages:

```
>enable
#debug ip mrouting
```

# debug ip ospf

Use the **debug ip ospf** command to activate debug messages associated with open shortest path first (OSPF) routing operations. Debug messages are displayed (real time) to the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages. Variations of this command include:

**debug ip ospf**
**debug ip ospf adj**
**debug ip ospf database-timer**
**debug ip ospf events**
**debug ip ospf flood**
**debug ip ospf hello**
**debug ip ospf lsa-generation**
**debug ip ospf packet**
**debug ip ospf retransmission**
**debug ip ospf spf**
**debug ip ospf tree**

*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

| | |
|---|---|
| **adj** | Displays OSPF adjacency events. |
| **database-timer** | Displays OSPF database timer. |
| **events** | Displays OSPF events. |
| **flood** | Displays OSPF flooding. |
| **hello** | Displays OSPF hello events. |
| **lsa-generation** | Displays OSPF link state advertisement (LSA) generation. |
| **packet** | Displays OSPF packets. |
| **retransmission** | Displays OSPF retransmission events. |
| **spf** | Displays OSPF shortest-path-first (SPF) calculations. |
| **tree** | Displays OSPF database tree. |

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Usage Examples

The following is an example of **debug ip ospf** command results:

>**enable**
**#debug ip ospf flood**
OSPF: Update LSA: id=c0a8020d rtid=192.168.2.13 area=11.0.0.0 type=1
OSPF: Update LSA: id=0b003202 rtid=11.0.50.2 area=11.0.0.0 type=1
OSPF: Queue delayed ACK lasid=0b003202 lsartid=11.0.50.2 nbr=11.0.50.2
OSPF: Rx ACK lasid=c0a8020d lsartid=192.168.2.13 nbr=11.0.50.2
OSPF: Received LSA ACK LSA_ID=-64.-88.2.13 LSA_RT_ID=-64.-88.2.13
OSPF: Rx ACK lasid=00000000 lsartid=192.168.2.13 nbr=11.0.50.2
OSPF: Received LSA ACK LSA_ID=0.0.0.0 LSA_RT_ID=-64.-88.2.13
OSPF: Sending delayed ACK
OSPF: Update LSA: id=c0a8020d rtid=192.168.2.13 area=11.0.0.0 type=1
OSPF: Flooding out last interface
OSPF: Update LSA: id=0b003202 rtid=11.0.50.2 area=11.0.0.0 type=1

# debug ip packet

Use the **debug ip packet** command to display debug messages associated with protocol-independent multicast (PIM) sparse assert transactions. Debug messages are displayed (real time) to the terminal (or Telnet) screen. Use the **no** form of this command to disable debug messages. Variations of this command include the following:

**debug ip packet**
**debug ip packet detail**
**debug ip packet dump**
**debug ip packet** <*name*>
**debug ip packet** <*name*> **detail**
**debug ip packet** <*name*> **dump**


*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

| | |
|---|---|
| **detail** | Optional. Displays IP packet detailed information. |
| **dump** | Optional. Displays IP packet detailed information on the console or Telnet terminal session.<br>Note: The console stream can be captured to a log file and used as an input file for display with ETHEREAL by using **text2pcap.exe**, which is a part of the ETHEREAL distribution.<br>Execute as follows: **text2pcap -l 101** <*input_file*> <*output_file*><br>Next, open the output file with ETHEREAL for display and decode. The typical lower layer information in ETHEREAL may not be present. This converted capture file is treated as a raw IP capture and also has no timestamp data. Remember to take advantage of access control lists (ACL's) to narrow down the amount of data being processed with this facility. This is a CPU intensive operation and also disables any fast flow/fast cache routing. |
| <*name*> | Optional. Specifies the name of the access control list. |

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following is sample output for the **debug ip packet** command:

>**enable**
**#debug ip packet**

IP: s= 192.168.8.101 (eth 0/1) d=192.168.7.2 (eth 0/2) g= 192.168.7.2, forward
IP: s= 192.168.7.2 (eth 0/2) d=192.168.8.101 (eth 0/1) g= 192.168.8.101, forward
IP: s= 192.168.8.101 (eth 0/1) d=192.168.7.2 (eth 0/2) g= 192.168.7.2, forward
IP: s= 192.168.7.2 (eth 0/2) d=192.168.8.101 (eth 0/1) g= 192.168.8.101, forward

Where:

**s=192.168.8.101 (eth 0/1)** indicates source address and interface of received packet.
**d=192.168.7.2 (eth 0/2)** indicates destination address and interface from which the packet is being sent.
**g=192.168.7.2** indicates the address of the next hop gateway.
**forward** indicates the router is forwarding this packet.

# debug ip pim-sparse

Use the **debug ip pim-sparse** command to display all protocol-independent multicast (PIM) sparse mode information. Debug messages are displayed (real time) to the terminal (or Telnet) screen. Use the **no** form of this command to disable debug messages.


*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

No subcommands.

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example activates all PIM sparse mode messages:

```
>enable
#debug ip pim-sparse
```

# debug ip pim-sparse assert

Use the **debug ip pim-sparse assert** command to display debug messages associated with protocol-independent multicast (PIM) sparse assert transactions. Debug messages are displayed (real time) to the terminal (or Telnet) screen. Use the **no** form of this command to disable debug messages. Variations of this command include:

**debug ip pim-sparse assert**
**debug ip pim-sparse assert event**
**debug ip pim-sparse assert event** <*multicast address*>
**debug ip pim-sparse assert state**
**debug ip pim-sparse assert state** <*multicast address*>

*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

| | |
|---|---|
| **event** | Optional. Displays PIM sparse assert events. |
| **state** | Optional. Displays PIM sparse assert state changes. |
| <*multicast address*> | Optional. Specifies multicast group IP address to filter. The multicast group IP address range is 244.0.0.0 to 239.255.255.255 or 224.0.0.0 /4. |

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example activates all PIM sparse assert event messages:

**>enable**
**#debug ip pim-sparse assert event**
14:25:05: PIMSM: Assert - MRoute (*, 239.255.255.250, eth 0/2) processed Received Join in NoInfo state
14:25:29: PIMSM: Assert - MRoute (10.100.13.240, 239.192.19.136, eth 0/2) processed Received Join in NoInfo state
14:25:29: PIMSM: Assert - MRoute (*, 239.192.19.136, eth 0/2) processed Received Join in NoInfo state
14:26:05: PIMSM: Assert - MRoute (*, 239.255.255.250, eth 0/2) processed Received Join in NoInfo state

# debug ip pim-sparse hello

Use the **debug ip pim-sparse hello** command to display protocol-independent multicast (PIM) sparse mode hello transactions. Debug messages are displayed (real time) to the terminal (or Telnet) screen. Use the **no** form of this command to disable debug messages.


*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

No subcommands.

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example activates PIM sparse mode hello messages:

```
>enable
#debug ip pim-sparse hello
```

# debug ip pim-sparse joinprune

Use the **debug ip pim-sparse joinprune** command to display protocol-independent multicast (PIM) sparse mode join and prune transactions. Debug messages are displayed (real time) to the terminal (or Telnet) screen. Use the **no** form of this command to disable debug messages. Variations of this command include:

**debug ip pim-sparse joinprune**
**debug ip pim-sparse joinprune event**
**debug ip pim-sparse joinprune event** <*multicast address*>
**debug ip pim-sparse joinprune state**
**debug ip pim-sparse joinprune state** <*multicast address*>

*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

| | |
|---|---|
| **event** | Optional. Displays PIM sparse join and prune events. |
| **state** | Optional. Displays PIM sparse join and prune state changes. |
| <*multicast address*> | Optional. Specifies multicast group IP address to filter. The multicast group IP address range is 244.0.0.0 to 239.255.255.255 or 224.0.0.0 /4. |

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example activates PIM sparse mode messages for all join and prune events and state changes:

**>enable**
**#debug ip pim-sparse joinprune**
14:27:05: PIMSM: Processed JOIN(*, 239.255.255.250) from 10.10.10.2
14:27:29: PIMSM: Processed JOIN(10.100.13.240, 239.192.19.136) from 10.10.10.2
14:27:29: PIMSM: Processed JOIN(*, 239.192.19.136) from 10.10.10.2
14:27:56: PIMSM: Sent JOIN(10.100.13.240, 239.192.19.136) to 10.100.13.240
14:28:05: PIMSM: Processed JOIN(*, 239.255.255.250) from 10.10.10.2

# debug ip pim-sparse packets

Use the **debug ip pim-sparse packets** command to display protocol-independent multicast (PIM) sparse mode packet information. Debug messages are displayed (real time) to the terminal (or Telnet) screen. Use the **no** form of this command to disable debug messages.Variations of this command include:

**debug ip pim-sparse packets**
**debug ip pim-sparse packets in**
**debug ip pim-sparse packets in** <*interface*>
**debug ip pim-sparse packets out**
**debug ip pim-sparse packets out** <*interface*>


*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

| | |
|---|---|
| **in** | Optional. Displays messages for inbound PIM sparse packets |
| **out** | Optional. Displays messages for outbound PIM sparse packets. |
| <*interface*> | Optional. Specifies an interface in the format <*interface type [slot/port | slot/port.sub-interface id | interface id | interface id.sub-interface id]*>. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **debug ip pim-sparse packets ?** for a list of valid interfaces. |

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example activates all PIM sparse packet messages (both inbound and outbound):

>**enable**
**#debug ip pim-sparse packets**

# debug ip pim-sparse register

Use the **debug ip pim-sparse register** command to display protocol-independent multicast (PIM) sparse source registration messages. Debug messages are displayed (real time) to the terminal (or Telnet) screen. Use the **no** form of this command to disable debug messages. Variations of this command include:

**debug ip pim-sparse register**
**debug ip pim-sparse register event**
**debug ip pim-sparse register event** <*multicast address*>
**debug ip pim-sparse register state**
**debug ip pim-sparse register state** <*multicast address*>


*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

| | |
|---|---|
| **event** | Optional. Displays PIM sparse register events. |
| **state** | Optional. Displays PIM sparse register state changes. |
| <*multicast address*> | Optional. Specifies multicast group IP address to filter. The multicast group IP address range is 244.0.0.0 to 239.255.255.255 or 224.0.0.0 /4. |

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example activates all PIM sparse registration changes:

**>enable**
**#debug ip pim-sparse register**
18:14:22: PIMSM: Registered new source (10.100.13.240, 239.192.19.136) from 10.10.10.1
18:14:22: PIMSM: RegisterStop(10.100.13.240, 239.192.19.136) sent to 10.10.10.1
18:14:53: PIMSM: RegisterStop(10.100.13.240, 239.192.19.136) sent to 10.10.10.1
18:16:17: PIMSM: RegisterStop(10.100.13.240, 239.192.19.136) sent to 10.10.10.1

# debug ip policy

Use the **debug ip policy** command to display policy-based routing events. Debug messages are displayed (real time) to the terminal (or Telnet) screen. Use the **no** form of this command to disable debug messages.


*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

No subcommands.

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example activates policy-based routing event messages:

```
>enable
#debug ip policy
```

# debug ip rip

Use the **debug ip rip** command to activate debug messages associated with Routing Information Protocol (RIP) operation in AOS. Debug messages are displayed (real time) to the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages. Variations of this command include:

**debug ip rip**
**debug ip rip events**

> *Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

| | |
|---|---|
| **events** | Optional. Displays only RIP protocol events. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

The **debug ip rip** command activates debug messages to provide information on RIP activity in AOS. RIP allows hosts and routers on a network to exchange information about routes.

## Usage Examples

The following example activates debug messages associated with RIP activity:

```
>enable
#debug ip rip
```

# debug ip routing

Use the **debug ip routing** command to activate debug messages associated with routing table events. Debug messages are displayed (real time) to the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages.


*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example activates debug messages associated with routing table events:

```
>enable
#debug ip routing
```

# debug ip tcp

Use the **debug ip tcp** command to activate debug messages associated with significant Transmission Control Protocol (TCP) events such as state changes, retransmissions, session aborts, etc., in AOS. Debug messages are displayed (real time) to the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages. Variations of this command include:

**debug ip tcp**
**debug ip tcp events**

*These debug events are logged for packets that are sent or received from the router. Forwarded TCP packets are not included.*

*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

| | |
|---|---|
| **events** | Optional. Displays only TCP protocol events. |

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 4.1 | Command was introduced. |

## Functional Notes

In the **debug ip tcp events** information, TCB stands for TCP task control block. The numbers which sometimes appear next to TCB (e.g., **TCB5** in the following example) represent the TCP session number. This allows you to differentiate debug messages for multiple TCP sessions.

## Usage Examples

The following is sample output for this command:

>**enable**
**#debug ip tcp events**
2003.02.17 07:40:56 IP.TCP EVENTS TCP: Allocating block 5
2003.02.17 07:40:56 IP.TCP EVENTS TCB5: state change: FREE->SYNRCVD
2003.02.17 07:40:56 IP.TCP EVENTS TCB5: new connection from 172.22.75.246:3473 to 10.200.2.201:23
2003.02.17 07:40:56 IP.TCP EVENTS TCB5: state change: SYNRCVD->ESTABLISHED [172.22.75.246:3473]
2003.02.17 07:41:06 IP.TCP EVENTS TCB5: Connection aborted -- error = RESET
2003.02.17 07:41:06 IP.TCP EVENTS TCB5: De-allocating tcb

# debug ip tcp md5

Use the **debug ip tcp md5** command to activate debug messages that detail the results of each incoming Transmission Control Protocol (TCP) packet's MD5 authentication with an internal route in AOS. Debug messages are displayed (real time) to the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages.


*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

No subcommands.

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Functional Notes

Debug messages will only be generated for TCP ports that have MD5 authentication enabled.

## Usage Examples

The following example activates debug messages associated with incoming TCP packet's MD5 authentication:

```
>enable
#debug ip tcp md5
```

# debug ip udp

Use the **debug ip udp** command to activate debug messages associated with User Datagram Protocol (UDP) send and receive events in AOS. Debug messages are displayed (real time) to the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages.


*These debug events are logged for packets that are sent or received from the router. Forwarded UDP packets are not included.*


*The overhead associated with this command takes up a large portion of your router's resources and at times can halt other router processes. It is best to only use the command during times when the network resources are in low demand (non-peak hours, weekends, etc.).*


*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

No subcommands.

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 4.1 | Command was introduced. |

## Functional Notes

In the **debug ip udp** information, the message **no listener** means that there is no service listening on this UDP port (i.e., the data is discarded).

## Usage Examples

The following is sample output for this command:

```
>enable
#debug ip udp
2003.02.17 07:38:48 IP.UDP RX: src=10.200.3.236:138, dst=10.200.255.255:138, 229 bytes, no listener
2003.02.17 07:38:48 IP.UDP RX: src=10.200.2.7:138, dst=10.200.255.255:138, 227 bytes, no listener
2003.02.17 07:38:48 IP.UDP RX: src=10.200.201.240:138, dst=10.200.255.255:138, 215 bytes, no
    listener
```

# debug ip urlfilter

Use the **debug ip urlfilter** command to display a summary of debug information for all URL filters being used. Debug messages are displayed (real time) to the terminal (or Telnet) screen. The verbose option gives more detailed information. Use the **no** form of this command to disable debug messages. Variations of this command include:

**debug ip urlfilter**
**debug ip urlfilter verbose**

*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

| | |
|---|---|
| **verbose** | Optional. Enables detailed debug messages. |

## Default Values

By default, all debug messages are disabled.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example shows the debug summary for all URL filters being used:

**>enable**
**#debug ip urlfilter**

2005.11.06 05:31:52 Connected to a Websense server
2005.11.06 05:33:26 Allowed http://www.adtran.com/

# debug isdn

Use the **debug isdn** command to activate debug messages associated with integrated services digital network (ISDN) events in AOS. Debug messages are displayed (real time) to the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages. Variations of this command include the following:

**debug isdn cc-ie**
**debug isdn cc-ie pri**
**debug isdn cc-ie pri** <*number*>
**debug isdn cc-messages**
**debug isdn cc-messages pri**
**debug isdn cc-messages pri** <*number*>
**debug isdn endpoint**
**debug isdn endpoint pri**
**debug isdn endpoint pri** <*number*>
**debug isdn interface**
**debug isdn interface pri**
**debug isdn interface pri** <*number*>
**debug isdn l2-formatted**
**debug isdn l2-formatted pri**
**debug isdn l2-formatted pri** <*number*>
**debug isdn l2-messages**
**debug isdn l2-messages pri**
**debug isdn l2-messages pri** <*number*>


*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

| | |
|---|---|
| **cc-ie** | Displays call control information elements. |
| **cc-messages** | Displays call control messages. |
| **endpoint** | Displays endpoint events. |
| **interface** | Displays ISDN interface events. |
| **l2-formatted** | Displays layer 2 formatted messages. |
| **l2-messages** | Displays layer 2 messages. |
| **pri** | Optional. Specifies the ISDN interface. |
| **pri** <*number*> | Optional. Specifies a specific ISDN interface. Range is 1 to 255. |

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example activates all layer 2 formatted messages:

```
>enable
#debug isdn l2-formatted
```

The following example activates layer 2 formatted messages received on ISDN interface PRI 1:

```
>enable
#debug isdn l2-formatted pri 1
```

# debug isdn group

Use the **debug isdn group** command to activate integrated services digital network (ISDN) group errors and messages. Debug messages are displayed (real time) to the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages. Variations of this command include:

**debug isdn group**
**debug isdn group** <*number*>

*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

| | |
|---|---|
| <*number*> | Optional. Specifies the ISDN group. Valid range is 1 to 255. |

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example activates debug messages for all ISDN groups:

>**enable**
**#debug isdn group**

# debug isdn resource-manager

Use the **debug isdn resource-manager** command to activate integrated services digital network (ISDN) resource manager errors and messages. Debug messages are displayed (real time) to the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages.


*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

No subcommands.

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example activates debug messages associated with the ISDN resource manager:

```
>enable
#debug isdn resource-manager
```

# debug isdn verbose

Use the **debug isdn verbose** command to activate all debug messages associated with integrated services digital network (ISDN) events in AOS. Debug messages are displayed (real time) to the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages.


*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

No subcommands.

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example activates all debug messages associated with ISDN activity:

```
>enable
#debug isdn verbose
```

# debug lldp

Use the **debug lldp** command to display debug output for all local loop demarkation point (LLDP) receive and transmit packets. Debug messages are displayed (real time) on the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages. Variations of this command include:

**debug lldp**
**debug lldp rx**
**debug lldp rx verbose**
**debug lldp tx**
**debug lldp tx verbose**
**debug lldp verbose**

*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

| | |
|---|---|
| **rx** | Optional. Shows information about received packets. |
| **tx** | Optional. Shows information about transmitted packets. |
| **verbose** | Optional. Shows detailed debugging information. |

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Usage Examples

The following example activates all possible debug messages associated with LLDP operation:

**>enable**
**#debug lldp rx**
**#debug lldp tx**
**#debug lldp verbose**

# debug port-auth

Use the **debug port-auth** command to generate debug messages used to aid in troubleshooting problems during the port authentication process. Debug messages are displayed (real time) on the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages. Variations of this command include the following:

**debug port-auth**
**debug port-auth auth-sm**
**debug port-auth bkend-sm**
**debug port-auth general**
**debug port-auth packet**
**debug port-auth packet [both | tx | rx]**
**debug port-auth reauth-sm**
**debug port-auth supp-sm**


*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

| | |
|---|---|
| **auth-sm** | Optional. Displays AuthPAE-state machine information. |
| **bkend-sm** | Optional. Displays backend-state machine information. |
| **general** | Optional. Displays configuration changes to the port authentication system. |
| **packet both** | Optional. Displays packet exchange information in both receive and transmit directions. |
| **packet rx** | Optional. Displays packet exchange information in the receive-only direction. |
| **packet tx** | Optional. Displays packet exchange information in the transmit-only direction. |
| **reauth-sm** | Optional. Displays reauthentication-state machine information. |
| **supp-sm** | Optional. Displays supplicant-state machine information. |

## Default Values

By default, all debug messages in the AOS are disabled.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |
| Release 10.1 | New options were introduced. |
| Release 13.1 | New options were introduced. |

## Usage Examples

The following example activates port authentication debug information on received packets:

>**enable**
**#debug port-auth packet rx**
Rcvd EAPOL Start for sess 1 on int eth 0/2

# debug port security

Use the **debug port security** command to display messages associated with port security. Debug messages are displayed (real time) to the terminal (or Telnet) screen. Use the **no** form of this command to disable debug messages.


*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

No subcommands.

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Usage Examples

The following example activates port security debug messages:

```
>enable
#debug port security
```

# debug ppp

Use the **debug ppp** command to activate debug messages associated with point-to-point protocol (PPP) operation in AOS. Debug messages are displayed (real time) to the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages. Variations of this command include:

**debug ppp authentication**
**debug ppp errors**
**debug ppp negotiation**
**debug ppp verbose**


*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

| | |
|---|---|
| **authentication** | Activates debug messages pertaining to PPP authentication (CHAP, PAP, EAP, etc.). |
| **errors** | Activates debug messages that indicate a PPP error was detected (mismatch in negotiation authentication, etc.). |
| **negotiation** | Activates debug messages associated with PPP negotiation. |
| **verbose** | Activates detailed debug messages for PPP operation. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

The **debug ppp** command activates debug messages to provide information on PPP activity in the system. PPP debug messages can be used to aid in troubleshooting PPP links.

## Usage Examples

The following example activates debug messages associated with PPP authentication activity:

```
>enable
#debug ppp authentication
```

# debug pppoe client

Use the **debug pppoe client** command to activate debug messages associated with point-to-point protocol over Ethernet (PPPoE) operation in AOS. Debug messages are displayed (real time) to the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages.


*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Usage Examples

The following example activates debug messages associated with PPPoE activity:

```
>enable
#debug pppoe client
```

# debug probe <*name*>

Use the **debug probe** command to activate debug messages associated with activities performed by the named probe object. Debug messages are displayed (real time) to the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages. Variations of this command include:

**debug probe**
**debug probe** <*name*>

## Syntax Description

| | |
|---|---|
| <*name*> | Optional. Specifies the probe object. |

## Default Values

By default, all debug messages in the AOS are disabled.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example activates debug messages associated with all probe objects:

>**enable**
**#debug probe**

The following example activates debug messages associated with the probe object named **probe_A**:

>**enable**
**#debug probe probe_A**

# debug radius

Use the **debug radius** command to enable debug messages from the RADIUS subsystem. Debug messages are displayed (real time) to the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages.

*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

No subcommands.

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

The **debug radius** messages show the communication process with the remote RADIUS servers.

## Usage Examples

The following is an example output for the **debug radius** command:

```
>enable
#debug radius
RADIUS AUTHENTICATION: Sending packet to 172.22.48.1 (1645).
RADIUS AUTHENTICATION: Received response from 172.22.48.1.
```

# debug sip

Use the **debug sip** command to activate debug messages associated with Session Initiation Protocol (SIP) events. Debug messages are displayed (real time) on the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages. Variations of this command include:

**debug sip cldu**
**debug sip location**
**debug sip manager**
**debug sip trunk-registration**
**debug sip trunk-registration** <*Txx*>
**debug sip trunk-registration** <*Txx*> <*identity*>
**debug sip user-registration**
**debug sip user-registration** <*extension*>

*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

| | |
|---|---|
| **cldu** | Optional. Activates SIP CLDU event debug messages. |
| **location** | Optional. Activates SIP location database event debug messages. |
| **manager** | Optional. Activates SIP stack manager event debug messages. |
| **proxy** | Optional. Activates SIP proxy event debug messages. |
| **proxy** <*subsource*> | Optional. Activates SIP proxy event debug messages for a specific subsource. |
| **trunk-registration** | Activates SIP trunk-registration event debug messages. |
| **trunk-registration** <*Txx*> | Optional. Activates SIP trunk-registration event debug messages for a specific trunk. For example: Txx (T01) where *xx* is the trunk's two-digit identifier. |
| **trunk-registration** <*Txx*> <*identity*> | Activates SIP trunk-registration event debug messages for a specific trunk. For example: Txx (T01) where *xx* is the trunk's two-digit identifier and <*identity*> is the specific name associated with the trunk. |
| **user-registration** | Activates SIP user-registration event debug messages. |
| **user-registration** <*extension*> | Optional. Activates SIP user-registration event debug messages for a specific trunk. |

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |
| Release 11.1 | Command was expanded to include proxy event messages. |

## Usage Examples

The following example activates all debug messages associated with SIP CLDU events:

>**enable**
**#debug sip cldu**

# debug sip stack

Use the **debug sip stack** command to activate debug messages associated with Session Initiation Protocol (SIP) stack events. Debug messages are displayed (real time) on the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages. Variations of this command include:

**debug sip stack debug**
**debug sip stack errors**
**debug sip stack exceptions**
**debug sip stack info**
**debug sip stack messages**
**debug sip stack messages summary**
**debug sip stack verbose**
**debug sip stack warnings**

*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

| | |
|---|---|
| **debug** | Activates SIP stack debug event debug messages. |
| **errors** | Activates SIP stack error event debug messages. |
| **exceptions** | Activates SIP stack exception event debug messages. |
| **info** | Activates SIP stack info event debug messages. |
| **messages** | Activates all SIP debug messages. |
| **messages summary** | Activates a summary of all SIP debug messages. |
| **verbose** | Activates all SIP stack event debug messages. |
| **warnings** | Activates SIP stack warning event debug messages. |

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example activates all debug messages associated with SIP stack events:

```
>enable
#debug sip stack
```

# debug sntp

Use the **debug sntp** command to enable debug messages associated with the Simple Network Time Protocol (SNTP). All SNTP packet exchanges and time decisions are displayed with these debugging events enabled. Debug messages are displayed (real time) to the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages. Variations of this command include the following:

**debug sntp**
**debug sntp client**
**debug sntp server**


*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

| | |
|---|---|
| **client** | Optional. Displays SNTP client information. |
| **server** | Optional. Displays SNTP server information |

## Default Values

By default, all debug messages in the AOS are disabled.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |
| Release 13.1 | Command was expanded to include the client and server options. |

## Functional Notes

The **debug sntp** command activates debug messages to aid in troubleshooting SNTP protocol issues.

## Usage Examples

The following is an example output for the **debug sntp** command:

>**enable**
**#debug sntp**
**#config term**
(config)**#sntp server timeserver.localdomain**
2002.12.11 15:06:37 SNTP.CLIENT sent Version 1 SNTP time request to 63.97.45.57
2002.12.11 15:06:37 SNTP.CLIENT received SNTP reply packet from 63.97.45.57
2002.12.11 15:06:37 SNTP.CLIENT setting time to 12-11-2002 15:06:02 UTC
2002.12.11 15:06:37 SNTP.CLIENT waiting for 86400 seconds for the next poll interval

# debug spanning-tree bpdu

Use the **debug spanning-tree bpdu** command to display bridge protocol data unit (BPDU) debug messages. When enabled, a debug message is displayed for each BPDU packet that is transmitted or received by the unit. Debug messages are displayed (real time) on the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages. Variations of this command include:

**debug spanning-tree bpdu all**
**debug spanning-tree bpdu receive**
**debug spanning-tree bpdu transmit**

*Turning on a large amount of debug information can adversely affect the performance of your unit.*

*Refer to debug spanning-tree on page 173 for more information.*

## Syntax Description

| | |
|---|---|
| **all** | Displays debug messages for BPDU packets that are transmitted and received by the unit. |
| **receive** | Displays debug messages for BPDU packets received by the unit. |
| **transmit** | Displays debug messages for BPDU packets transmitted by the unit. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following example displays debug messages for BPDU packets that are transmitted and received by the unit:

```
>enable
#debug spanning-tree bpdu all
```

# debug spanning-tree

Use the **debug spanning-tree** command to enable the display of spanning-tree debug messages. Debug messages are displayed (real time) on the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages. Variations of this command include:

**debug spanning-tree config**
**debug spanning-tree events**
**debug spanning-tree general**
**debug spanning-tree topology**


*Turning on a large amount of debug information can adversely affect the performance of your unit.*


*Refer to debug spanning-tree bpdu on page 172 for more information.*

## Syntax Description

| | |
|---|---|
| **config** | Enables the display of spanning-tree debug messages when configuration changes occur. |
| **events** | Enables the display of debug messages when spanning-tree protocol events occur. |
| **general** | Enables the display of general spanning-tree debug messages. |
| **topology** | Enables the display of debug messages when spanning-tree protocol topology events occur. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |
| Release 12.1 | Command was expanded to include **topology**. |

## Usage Examples

The following example enables the display of general spanning-tree debug messages:

```
>enable
#debug spanning-tree general
```

# debug stack

Use the **debug stack** command to enable switch-stacking debug messages. Debug messages are displayed (real time) on the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages.Variations of this command include:

**debug stack**
**debug stack switch**
**debug stack verbose**


*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

| | |
|---|---|
| **switch** | Optional. Enables messages specific to the stack ports (stack switch API information). |
| **verbose** | Optional. Enables detailed messages specific to the stack protocol. |

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Usage Examples

The following example activates the possible debug stack messages:

**>enable**
**#debug stack switch**
**#debug stack verbose**

# debug system

Use the **debug system** command to enable debug messages associated with system events (i.e., login, logouts, etc.). Debug messages are displayed (real time) to the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages.


*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

No subcommands.

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Usage Examples

The following example activates debug messages associated with system information:

```
>enable
#debug system
```

# debug tacacs+

Use the **debug tacas+** command to activate debug messages associated with terminal access controller access control system (TACACS+) protocol. Debug messages are displayed (real time) to the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages. Variations of this command include:

**debug tacacs+**
**debug tacacs+ events**
**debug tacacs+ packets**


*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

| | |
|---|---|
| **events** | Optional. Activates TACACS+ event debug messages. |
| **packets** | Optional. Activates TACACS+ packet debug messages. |

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example activates debug messages associated with the TACACS+ protocol:

```
>enable
#debug tacacs+ packets
```

# debug tftp

Use the **debug tftp packets** command to activate debug messages associated with Trivial File Transfer Protocol (TFTP) packets. Debug messages are displayed (real time) to the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages. Variations of this command include:

**debug tftp client packets**
**debug tftp server events**
**debug tftp server packets**


*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

| | |
|---|---|
| **client packets** | Activates TFTP client packet debug messages. |
| **server events** | Activates TFTP server event debug messages. |
| **server packets** | Activates TFTP server packet debug messages. |

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example activates debug messages associated with TFTP server packets:

**>enable**
**#debug tftp server packets**

# debug track <*name*>

Use the **debug track** command to activate debug messages associated with activities performed by the named track object. Debug messages are displayed (real time) to the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages. Variations of this command include:

**debug track**
**debug track** <*name*>

## Syntax Description

| | |
|---|---|
| <*name*> | Specifies the track object. |

## Default Values

By default, all debug messages in the AOS are disabled.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example activates debug messages associated with all track objects:

>**enable**
**#debug track**

The following example activates debug messages associated with the track object named **track_1**:

>**enable**
**#debug track track_1**

## debug voice

Use the **debug voice** command to activate debug messages associated with voice functionality. Debug messages are displayed (real time) on the terminal (or Telnet) screen. Use the **no** form of this command to disable the debug messages. Variations of this command include the following:

**debug voice autoattendant**
**debug voice mail**
**debug voice mail** <*subsource*>
**debug voice phonemanager**
**debug voice phonemanager** <*slot/port*>
**debug voice promptstudio**
**debug voice proxydial**
**debug voice rtp channel**
**debug voice rtp manager**
**debug voice rtp provider**
**debug voice rtp verbose**
**debug voice smdr**
**debug voice smdr** <*number*>
**debug voice stationaccount**
**debug voice stationaccount** <*station*>
**debug voice summary**
**debug voice switchboard**
**debug voice switchboard** <*subsource*>
**debug voice switchboard call**
**debug voice switchboard call** <*subsource*>
**debug voice switchboard ccm**
**debug voice toneservices**
**debug voice toneservices** <*interface*>
**debug voice toneservices** <*interface*> <*slot/port*>
**debug voice trunkaccount**
**debug voice trunkaccount** <*trunk*>
**debug voice trunkaccount** <*trunk*> <*appearance*>
**debug voice trunkmanager**
**debug voice trunkmanager** <*trunk*>
**debug voice trunkport**
**debug voice trunkport** <*slot/port*>
**debug voice verbose**


*Turning on a large amount of debug information can adversely affect the performance of your unit.*

## Syntax Description

| | |
|---|---|
| **autoattendant** | Activates auto-attendant event debug messages. |
| **mail** | Activates voice mail event debug messages. |
| **mail** <*subsource*> | Optional. Activates voice mail event debug messages for a specific subsource. |
| **phonemanager** | Activates phone manager event debug messages. |
| **phonemanager** <*slot/port*> | Optional. Activates phone manager event debug messages for a specific slot/port. |
| **promptstudio** | Activates prompt-studio event debug messages. |
| **proxydial** | Activates proxy dial event debug messages |
| **rtp channel** | Activates RTP channel event debug messages. |
| **rtp manager** | Activates RTP manager event debug messages. |
| **rtp provider** | Activates RTP provider event debug messages. |
| **rtp verbose** | Activates detailed RTP debug messages. |
| **smdr** | Activates SMDR event debug messages. |
| **smdr** <*number*> | Optional. Activates SMDR event debug messages for a specific to or from number.. event debug messages for a specific station. |
| **summary** | Activates simple voice event debug messages. |
| **switchboard** | Activates switchboard event debug messages. |
| **switchboard** <*subsource*> | Optional. Activates switchboard event debug messages for a specific subsource. |
| **switchboard call** | Activates switchboard call state machine event debug messages. |
| **switchboard call** <*subsource*> | Optional. Activates switchboard call state machine event debug messages for a specific subsource. |
| **switchboard ccm** | Activates switchboard call connection manager event debug messages. |
| **toneservices** | Activates debug messages associated with tone service events. |
| <*interface*> | Specifies an interface type in the format <*interface type*>. For example, for a FXS interface use **fxs**. |
| <*slot/port*> | Specifies an individual port to debug within an interface type in the format <*slot/port*>. For example, for an individual FXS port use **fxs 0/1**. |
| **trunkaccount** | Activates trunk account event debug messages. |
| **trunkaccount** <*trunk*> | Optional. Activates trunk account event debug messages for a specific trunk. |
| <*appearance*> | Optional. Specifies specific trunk appearance. |
| **trunkmanager** | Activates trunk manager event debug messages. |
| **trunkmanager** <*trunk*> | Optional. Activates trunk manager event debug messages for a specific trunk. |
| **trunkport** | Activates trunkport event debug messages. |
| **trunkport** <*slot/port*> | Optional. Activates trunkport event debug messages for a specific slot/port. |

| | |
|---|---|
| **verbose** | Optional. Displays the entire running configuration to the terminal screen (versus only the non-default values). |

## Default Values

By default, all debug messages in AOS are disabled.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |
| Release 10.1 | Command was expanded to include **toneservices**. |
| Release 12.1 | Command was expanded to include **autoattendant**, **mail**, and **switchboard call**. |
| Release 13.1 | Command was expanded to include more options. |

## Usage Examples

The following example activates all debug messages associated with voice functionality:

>**enable**
**#debug voice**

# dir

Use the **dir** command to display a directory list of all files on the system in flash memory or on the installed compact flash card or all files on the system in flash memory or on the installed compact flash card matching the specified pattern. Variations of this command include:

**dir**
**dir** *<pattern>*
**dir cflash**
**dir cflash** *<pattern>*
**dir flash**
**dir flash** *<pattern>*

## Syntax Description

| | |
|---|---|
| *<pattern>* | Lists all files stored in flash that match the specified pattern. When a wildcard (*) is specified, only files located in the specified location matching the listed pattern are displayed. For example, ***.biz** displays all files with the .biz extension.<br>When no wildcard is specified, the entire contents of flash memory is displayed. |
| **cflash** | Specifies files located on the installed compact flash card. |
| **flash** | Specifies files located on the system in flash memory. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |
| Release 12.1 | Command expanded to include compact flash. |

## Usage Examples

The following is sample output from the **dir** command:

```
>enable
#dir flash
3563529 NV2100A-10-05-00-E.biz
   2438 startup-config
   2484 startup-config.bak
3694712 bytes used, 3007368 available, 6702080 total
```

# disable

Use the **disable** command to exit the Enable mode and enter the Basic mode.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example exits the Enable mode and enters the Basic Command mode:

```
#disable
>
```

# enable

Use the **enable** command to enter a password for the Enable mode.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Functional Notes

The Enable Command mode provides access to operating and configuration parameters and should be password protected to prevent unauthorized use. Use the **enable password** command (found in the Global Configuration mode) to specify an Enable Command mode password. If the password is set, access to the Enable Commands (and all other “privileged” commands) is only granted when the correct password is entered. Refer to *enable password <password>* on page 479 for more information.

## Usage Examples

The following example enters the Enable Command mode and defines an Enable Command mode password:

```
>enable
Password: *****
#
```

# erase

Use the **erase** command to erase the specified file from the system flash memory or an installed compact flash card. Variations of this command include:

**erase** <*filename*>
**erase cflash** <*filename*>
**erase dynvoice-config**
**erase flash** <*filename*>
**erase startup-config**

## Syntax Description

| | |
|---|---|
| <*filename*> | Specifies the name of the file to erase. The asterisk (*) can be used as a wildcard to specify a pattern for erasing multiple files. When a wildcard is specified, only files matching the listed pattern are erased. |
| **cflash** | Specifies the location of the file to erase as the installed compact flash card. |
| **dynvoice-config** | Erases the dynvoice-config file stored in the flash memory. |
| **flash** | Specifies the location of the file to erase as the system flash memory. |
| **startup-config** | Erases the startup configuration file stored in flash memory. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 12.1 | Command was expanded to include the dynvoice-config option. |

## Usage Examples

The following example erases the startup configuration file stored in flash memory:

```
>enable
#erase startup-config
```

If a new startup-configuration file is not specified before power-cycling the unit, AOS will initialize using a blank configuration.

# events

Use the **events** command to enable event reporting to the current command line interface (CLI) session. Use the **no** form of this command to disable all event reporting to the current CLI session.

## Syntax Description

No subcommands.

## Default Values

By default, this command is enabled.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Usage Examples

The following example enables event reporting:

```
>enable
#events
```

# exception report generate

Use the **exception report generate** command to immediately generate an exception report.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example immediately generates an exception report:

```
>enable
#exception report generate
```

# factory-default

Use the **factory-default** command to reset the unit to the factory default settings.


*Performing an AOS **factory-default** disrupts data traffic.*

## Syntax Description

No subcommands.

## Default Values

No defaults necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

After you issue this command, the system responds by first warning you that restoring the factory default settings will erase the current configurations. It then asks if you would like to proceed. Choose **n** to return to the command prompt (no configuration changes are made). Choose **y** to erase the startup-configuration, replace it with the factory-default configuration, and reboot the unit. After reboot, the new configuration takes effect.

## Usage Examples

The following example resets the unit to factory default settings:

>**enable**
**#factory-default**

WARNING - Restoring the factory default settings will erase the current startup and running configurations and will reboot the unit.

Restore factory default settings?[y/n]**y**

Startup configuration written.

Rebooting the system. Please wait...

# logout

Use the **logout** command to terminate the current session and return to the login screen.

## Syntax Description

No subcommands.

## Default Values

No defaults necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example shows the logout command being executed in Enable mode:

```
>enable
#logout
Session now available
Press RETURN to get started.
```

# ping

Use the **ping** command (at the Enable mode prompt) to verify IP network connectivity. Variations of this command include:

**ping**
**ping** <*ip address*>
**ping** <*ip address*> **data** <*string*>
**ping** <*ip address*> **repeat** <*number*>
**ping** <*ip address*> **size** <*value*>
**ping** <*ip address*> **source** <*ip address*>
**ping** <*ip address*> **timeout** <*value*>

## Syntax Description

| | |
|---|---|
| <*ip address*> | Optional. Specifies the IP address of the system to ping. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**).<br>Entering the **ping** command with no specified IP address prompts the user with parameters for a more detailed **ping** configuration. Refer to *Functional Notes* (below) for more information. |
| **data** <*string*> | Specifies an alphanumerical string to use (the ASCII equivalent) as the data pattern in the ECHO_REQ packets. |
| **repeat** <*number*> | Specifies the number of ping packets to send to the system. Valid range is 1 to 1,000,000. |
| **size** <*value*> | Specifies the datagram size (in bytes) of the ping packet. Valid range is 1 to 1448 bytes. |
| **source** <*ip address*> | Specifies the IP address to use as the source address in the ECHO_REQ (or interface) packets. |
| **timeout** <*value*> | Specifies the timeout period after which the ping is considered unsuccessful. Valid range is 1 to 5 seconds. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

The **ping** command helps diagnose basic IP network connectivity using the Packet Internet Groper program to repeatedly bounce Internet Control Message Protocol (ICMP) Echo_Request packets off a system (using a specified IP address). AOS allows executing a standard **ping** request to a specified IP address or provides a set of prompts to configure a more specific **ping** configuration.

The following is a list of output messages from the **ping** command:

| | |
|---|---|
| ! | Success |
| - | Destination Host Unreachable |
| $ | Invalid Host Address |
| X | TTL Expired in Transit |
| ? | Unknown Host |
| * | Request Timed Out |

The following is a list of available extended **ping** fields with descriptions:

| | |
|---|---|
| Extended Commands | Specifies whether additional commands are desired for more ping configuration parameters. Answer yes (**y**) or no (**n**). |
| Source Address | Specifies the IP address to use as the source address in the ECHO_REQ (or interface) packets. |
| Data Pattern | Specifies an alphanumerical string to use (the ASCII equivalent) as the data pattern in the ECHO_REQ packets. |
| Sweep Range of Sizes | Varies the sizes of the ECHO_REQ packets transmitted. |
| Sweep Min Size | Specifies the minimum size of the ECHO_REQ packet. Valid range is 0 to 1488. |
| Sweep Max Size | Specifies the maximum size of the ECHO_REQ packet. Valid range is Sweep Min Size to 1448. |
| Sweep Interval | Specifies the interval used to determine packet size when performing the sweep. Valid range is 1 to 1448. |
| Verbose Output | Specifies an extended results output. |

## Usage Examples

The following is an example of a successful **ping** command:

>**enable**
**#ping**
Target IP address:**192.168.0.30**
Repeat count[1-1000000]:**5**
Datagram Size [1-1000000]:**100**
Timeout in seconds [1-5]:**2**
Extended Commands? [y or n]:**n**
Type CTRL+C to abort.
Legend: '!' = Success '?' = Unknown host '$' = Invalid host address
'*' = Request timed out '-' = Destination host unreachable
'x' = TTL expired in transit

Pinging 192.168.0.30 with 100 bytes of data:
!!!!!
Success rate is 100 percent (5/5) round-trip min/avg/max = 19/20.8/25 ms

# ping stack-member *<number>*

Use the **ping stack-member** command to ping a member of the stack.

## Syntax Description

| | |
|---|---|
| *<number>* | Specified which member of the stack to ping. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Functional Notes

This command is available only in stack-master mode.

## Usage Examples

The following example pings a member of the stack:

```
>enable
#ping stack-member 3
Type CTRL+C to abort.
Legend: '!' = Success, '?' = Unknown host, '$' = Invalid host address
      '*' = Request timed out, '-' = Destination host unreachable
      'x' = TTL expired in transit
Sending 5, 100-byte ICMP Echos to 169.254.0.3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/2.2/3 ms
#
```

# reload

Use the **reload** command to preform a manual reload of AOS. Variations of this command include:

**reload**
**reload cancel**
**reload in** <*delay*>


*Performing an AOS **reload** disrupts data traffic.*

## Syntax Description

| | |
|---|---|
| **cancel** | Optional. Deactivates a pending **reload** command. |
| **in** <*delay*> | Optional. Specifies a delay period in minutes (mmm) or hours and minutes (hh:mm) that AOS will wait before reloading. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example reloads the AOS software in 3 hours and 27 minutes:

**>enable**
**#reload in 03:27**

The following example reloads the AOS software in 15 minutes:

**>enable**
**#reload in 15**

The following example terminates a pending reload command:

**>enable**
**#reload cancel**

# show access-lists

Use the **show access-lists** command to display all configured access control lists in the system (or a specific list). Variations of this command include:

**show access-lists**
**show access-lists** <*name*>

## Syntax Description

| | |
|---|---|
| <*name*> | Optional. Specifies a particular access control list to display. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |

## Functional Notes

The **show access-lists** command displays all configured access control lists in the system. All entries in the access control list are displayed, and a counter indicating the number of packets matching the entry is listed.

## Usage Examples

The following is a sample output from the **show access-lists** command:

>**enable**
**#show access-lists**
Standard access list MatchAll
permit host 10.3.50.6 (0 matches)
permit 10.200.5.0 wildcard bits 0.0.0.255 (0 matches)
extended access list UnTrusted
deny icmp 10.5.60.0 wildcard bits 0.0.0.255 any source-quench (0 matches)
deny tcp any (0 matches)

# show arp

Use the **show arp** command to display the Address Resolution Protocol (ARP) table. Variations of this command include:

**show arp**
**show arp realtime**

> *Using the **realtime** argument for this command can adversely affect the performance or your unit.*

## Syntax Description

| | |
|---|---|
| **realtime** | Optional. Displays full-screen output in real time. Refer to the *Functional Notes* below for more information. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 10.1 | The real time display option was introduced. |

## Functional Notes

Use the **realtime** argument for this command to display full-screen output in real time. Information is continuously updated on the console until you either freeze the data (by pressing the **F** key) or exit **realtime** mode (by pressing **Ctrl-C**). If there is not enough room on the screen for all available data, the information will truncate at the bottom of the screen. In order to maximize the amount of data displayed, increase the terminal length (using the **terminal length** command; refer to *terminal length <number>* on page 410).

## Usage Examples

The following is a sample output of the **show arp** command:

```
>enable
#show arp
ADDRESS         TTL (min)   MAC ADDRESS        LAST UPDATED (min)   INTERFACE
192.168.30.36   13          00:E0:7D:88:1A:B9  4260                 eth 0/1
192.168.30.253  17          02:60:8C:DD:0A:CE  4264                 eth 0/1
224.0.0.9       71578541    01:00:5E:00:00:09  0                    eth 0/2
```

# show atm pvc

Use the **show atm** command to display information specific to the asynchronous transfer mode (ATM) interface's permanent virtual circuit (PVC). Variations of this command include the following:

**show atm pvc**
**show atm pvc interfaces atm** <*interface*>

## Syntax Description

| | |
|---|---|
| **interfaces atm** <*interface*> | Displays the ATM PVC information for a specific PVC. Specify an ATM interface (valid range is 1 to 1023) or a sub-interface in the format <*interface id.sub-interface id*>. For example, **1.1**. Using this command without specifying an interface will display all information all ATM PVCs. |

## Default Values

No default is necessary for this command.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Usage Examples

The following is sample output from the **show atm pvc interfaces** command:

```
>enable
#show atm pvc interface atm 1.1
Name     VPI  VCI  Encap Type  SC   Peak Kbps  Avg/Min Kbps  Burst Cells  Status
atm 1.1  0    200  SNAP        N/A  0          0             0            Active
```

# show atm traffic interface atm <*interface*>

Use the **show atm traffic** command to display traffic information specific to the asynchronous transfer mode (ATM) interface.

## Syntax Description

| | |
|---|---|
| <*interface*> | Specifies an ATM port number. Specify an ATM interface (valid range is 1 to 1023) or a sub-interface in the format <*interface id.sub-interface id*>. For example, **1.1**. |

## Default Values

No default is necessary for this command.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Usage Examples

The following is sample output from the **show atm traffic** command from ATM interface 1:

```
>enable
#show atm traffic interface atm 1
atm 1 is UP, line protocol is UP
 BW 896 Kbit/s
 16 maximum active VCCs, 16 VCCs per VP, 1 current VCCs
Queueing strategy: Per VC Queueing
 5 minute input rate 32 bits/sec, 0 packets/sec
 5 minute output rate 0 bits/sec, 0 packets/sec
    19 packets input, 1357 bytes
    0 pkts discarded, 0 error pkts, 0 unknown protocol pkts
    45 cells received, 0 OAM cells received
    0 packets output, 0 bytes
    0 tx pkts discarded, 0 tx error pkts 0 internal tx error pkts
    0 cells sent, 0 OAM cells sent
```

The following is sample output from the **show atm traffic** command from ATM sub-interface 1.1:

```
#show atm traffic interface atm 1.1
 27 Input Packets
 0 Output Packets
 72 Cells received, 0 OAM cells received
 F5 InEndLoopReq: 0 F5 InEndLoopResp: 0 F5 InAIS: 0 F5 InRDI: 0
 0 Cells sent, 0 OAM cells sent
 F5 OutEndLoopReq: 0 F5 OutEndLoopResp: 0 F5 OutAIS: 0 F5 OutRDI: 0
 0 OAM Loopback Successes 0 OAM Loopback Failures
```

# show auto-config

Use the **show auto-config** command to display auto-configuration status.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following is a sample output of the **show auto-config** command:

```
>enable
#show auto-config
Auto-Config is enabled, current status: Done.
 TFTP Server is 10.20.20.1
 Config filename is 1524STfile
      Maximum retry count is 0 (repeat indefinitely), total retries is 0
```

# show bridge

Use the **show bridge** command to display a list of all configured bridge groups (including individual members of each group). Enter an interface or a bridge number to display the corresponding list. Variations of this command include:

**show bridge**
**show bridge** <*number*>
**show bridge** <*interface*>

## Syntax Description

| | |
|---|---|
| <*interface*> | Optional. Displays all bridge groups associated with the specific interface. Specify an interface in the format *<interface type [slot/port \| slot/port.sub-interface id \| interface id \| interface id.sub-interface id]>*. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type the **show bridge ?** command to display a list of applicable interfaces. |
| <*number*> | Optional. Displays a specific bridge group. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 9.1 | Command was expanded to include HDLC interface. |

## Usage Examples

The following is a sample output from the **show bridge** command:

```
>enable
#show bridge
Total of 300 station blocks 295 free
```

| Address | Action | Interface | Age | Rx Count | Tx Count |
|---|---|---|---|---|---|
| 00:04:51:57:4D:5A | forward | eth 0/1 | 0 | 7133392 | 7042770 |
| 00:04:5A:57:4F:2A | forward | eth 0/1 | 0 | 402365 | 311642 |
| 00:10:A4:B3:A2:72 | forward | eth 0/1 | 4 | 2 | 0 |
| 00:A0:C8:00:8F:98 | forward | eth 0/1 | 0 | 412367 | 231 |
| 00:E0:81:10:FF:CE | forward | fr 1.17 | 0 | 1502106 | 1486963 |

# show buffers

Use the **show buffers** command to display the statistics for the buffer pools on the network server. Variations of this command include:

**show buffers**
**show buffers realtime**

> NOTE *Using the **realtime** argument for this command can adversely affect the performance or your unit.*

## Syntax Description

| | |
|---|---|
| **realtime** | Optional. Displays full-screen output in real time. Refer to the *Functional Notes* below for more information. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Functional Notes

Use the **realtime** argument for this command to display full-screen output in real time. Information is continuously updated on the console until you either freeze the data (by pressing the **F** key) or exit **realtime** mode (by pressing **Ctrl-C**). If there is not enough room on the screen for all available data, the information will truncate at the bottom of the screen. In order to maximize the amount of data displayed, increase the terminal length (using the **terminal length** command; refer to *terminal length <number>* on page 410).

## Usage Examples

The following is a sample output from the **show buffers** command:

```
>enable
#show buffers
Buffer handles: 119 of 2000 used.
```

| Pool | Size | Total | Used | Available | Max. Used |
|---|---|---|---|---|---|
| 0 | 1800 | 1894 | 119 | 1775 | 122 |
| 1 | 2048 | 64 | 0 | 64 | 0 |
| 2 | 4096 | 32 | 0 | 32 | 0 |
| 3 | 8192 | 4 | 0 | 4 | 0 |

# show buffers users

Use the **show buffers users** command to display a list of the top users of packet buffers. Typically, this command will only be used as a debug tool by ADTRAN personnel. Variations of this command include:

**show buffers users**
**show buffers users realtime**


*Using the **realtime** argument for this command can adversely affect the performance or your unit.*

## Syntax Description

| | |
|---|---|
| **realtime** | Optional. Displays full-screen output in real time. Refer to the *Functional Notes* below for more information. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Functional Notes

Use the **realtime** argument for this command to display full-screen output in real time. Information is continuously updated on the console until you either freeze the data (by pressing the **F** key) or exit **realtime** mode (by pressing **Ctrl-C**). If there is not enough room on the screen for all available data, the information will truncate at the bottom of the screen. In order to maximize the amount of data displayed, increase the terminal length (using the **terminal length** command; refer to *terminal length <number>* on page 410).

## Usage Examples

The following is a sample from the **show buffers users** command:

>**enable**
**#show buffers users**
Number of users: 7

| Ran | User | Count |
|---|---|---|
| 1 | 0x0052f4f8 | 59 |
| 2 | 0x0051a4fc | 32 |
| 3 | 0x00528564 | 8 |
| 4 | 0x0053c1c8 | 7 |
| 5 | fixedsize | 5 |
| 6 | 0x001d8298 | 2 |
| 7 | 0x0010d970 | 1 |
| 8 | 0x00000000 | 0 |
| 9 | 0x00000000 | 0 |
| 10 | 0x00000000 | 0 |
| 11 | 0x00000000 | 0 |

# show cflash

Use the **show cflash** command to display a list of all files currently stored in compact flash memory or details about a specific file stored in compact flash memory. Variations of this command include:

**show cflash**
**show cflash** *<filename>*

## Syntax Description

| | |
|---|---|
| *<filename>* | Optional. Displays details for a specified file located in flash memory. Enter a wildcard (such as ***.biz**) to display the details for all files matching the entered pattern. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following is a sample **show cflash** output:

```
>enable
#show cflash
(dir)      0 SystemDefaultPrompts
(dir)      0 VoiceMail
9377163 NV7100A-12-00-23-E.biz
11110890 sip.ld
8767439 NV7100A-11-03-02-E.biz
8771176 NV7100A-11-03-02d-E.biz
8773148 NV7100A-11-03-03-E.biz
48508928 bytes used, 207319040 available, 255827968 totall
```

# show channel-group

Use the **show channel-group** command to display detailed information regarding port aggregation of a specified channel group (i.e., channel groups and their associated ports). Variations of this command include the following:

**show channel-group port-channel load-balance**
**show channel-group summary**
**show channel-group** <*number*> **summary**

## Syntax Description

| | |
|---|---|
| **port-channel load-balance** | Displays the current load-balance scheme. |
| **summary** | Summarizes the state of all channel groups or of a specific channel group (if specified by the <*number*> argument). |
| <*number*> | Optional. Specifies the channel group using the channel group ID (16). |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following is a sample from the **show channel-group** command:

```
>enable
#show channel-group summary
Channel Group       Port channel        Associated Ports
--------------------  -----------------   ----------------------
1                   1                   eth 0/2 eth 0/3
2                   2                   eth 0/5 eth 0/6 eth 0/7
```

# show clock

Use the **show clock** command to display the system time and date entered using the **clock set** command. Refer to the section *clock set* <*time*> <*day*> <*month*> <*year*> on page 90 for more information. Variations of this command include:

**show clock**
**show clock detail**

## Syntax Description

| | |
|---|---|
| **detail** | Optional. Displays more detailed clock information, including the time source. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example displays the current time and data from the system clock:

>**show clock**
23:35:07 UTC Tue Aug 20 2002

# show configuration

Use the **show configuration** command to display a text printout of the startup configuration file stored in nonvolatile random access memory (NVRAM).

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following is a sample output of the **show configuration** command:

```
>enable
#show configuration
!
!
no enable password
!
ip subnet-zero
ip classless
ip routing
!
event-history on
no logging forwarding
logging forwarding priority-level info
no logging email
!
ip policy-timeout tcp all-ports 600
ip policy-timeout udp all-ports 60
ip policy-timeout icmp 60
!
!
!
interface eth 0/1
speed auto
  no ip address
  shutdown
!
interface dds 1/1
```

```
  shutdown
!
interface bri 1/2
  shutdown
!
!
ip access-list standard Outbound
  permit host 10.3.50.6
  permit 10.200.5.0 0.0.0.255
!
!
ip access-list extended UnTrusted
  deny   icmp 10.5.60.0 0.0.0.255 any source-quench
  deny   tcp any any
!
no ip snmp agent
!
!
!
line con 0
  no login
!
line telnet 0
  login
line telnet 1
  login
line telnet 2
  login
line telnet 3
  login
line telnet 4
  login
!
```

# show connections

Use the **show connections** command to display information (including TDM group assignments) for all active connections.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |

## Usage Examples

The following is sample output from the **show connections** command:

```
>enable
#show connections
Displaying all connections....
```

| Conn ID | From | To |
|---|---|---|
| 1 | atm 1 | adsl 1/1 |
| 2 | ppp 1 | t1 2/1, tdm-group 1 |
| 3 | ppp 1 | t1 2/2, tdm-group 1 |
| 4 | ppp 3 | e1 3/1, tdm-group 1 |
| 5 | ppp 3 | e1 3/2, tdm-group 1 |
| 6 | ppp 3 | e1 3/3, tdm-group 1 |

# show crypto ca

Use the **show crypto ca** command to display information regarding certificates and profiles. Variations of this command include:

**show crypto ca certificates**
**show crypto ca crls**
**show crypto ca profiles**

## Syntax Description

| | |
|---|---|
| **certificates** | Displays information on all certificates. |
| **crls** | Displays a summary of all certificate revocation lists (CRLs) for each CA. |
| **profiles** | Displays information on all configured CA profiles. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced (enhanced software version only). |

## Usage Examples

The following is a sample from the **show crypto ca certificates** command:

```
>enable
#show crypto ca certificates
CA Certificate
 Status: Available
 Certificate Serial Number: 012d
 Subject Name: /C=FI/O=SSH Communications Security/OU=Web test/CN=Test CA 1
 Issuer: /C=FI/O=SSH Communications Security/OU=Web test/CN=Test CA 1
 CRL Dist. Pt: /C=FI/O=SSH Communications Security/OU=Web test/CN=Test CA 1
 Start date is Jan 9 16:25:15 2003 GMT
 End date is Dec 31 23:59:59 2003 GMT
 Key Usage:
   Non-Repudiation
   Key Encipherment
   Data Encipherment
   CRL Signature
   Encipherment Only
```

# show crypto ike

Use the **show crypto ike** command to display information regarding the IKE configuration. Variations of this command include the following:

**show crypto ike client configuration pool**
**show crypto ike client configuration pool** <*name*>
**show crypto ike policy**
**show crypto ike policy** <*value*>
**show crypto ike remote-id** <*remote-id*>
**show crypto ike sa**

## Syntax Description

| | |
|---|---|
| **client configuration pool** | Displays the list of all configured IKE client configuration pools. |
| <*name*> | Optional. Displays detailed information regarding the specified IKE client configuration pool. |
| **policy** | Displays information on all IKE policies. Indicates if client configuration is enabled for the IKE policies and displays the pool names. |
| <*value*> | Optional. Displays detailed information on the specified IKE policy. This number is assigned using the **crypto ike policy** command. Refer to *crypto ike* on page 467 for more information. |
| **remote-id** <*remote-id*> | Displays information on all IKE information regarding the remote-id. The remote-id value is specified using the **crypto ike remote-id** command (refer to *crypto ike remote-id* on page 471). |
| **sa** | Displays the configuration of active IKE security associations. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 4.1 | Command was introduced. |

## Usage Examples

The following is a sample from the **show crypto ike policy** command:

```
>enable
#show crypto ike policy
Crypto IKE Policy 100
 Main mode
 Using System Local ID Address
 Peers:
  63.105.15.129
initiate main
respond anymode
 Attributes:
    10
      Encryption: 3DES
      Hash: SHA
      Authentication: Pre-share
      Group: 1
      Lifetime: 900 seconds
```

# show crypto ipsec

Use the **show crypto ipsec** command to display information regarding the IPSec configuration. Variations of this command include the following:

**show crypto ipsec sa**
**show crypto ipsec sa address** <*ip address*>
**show crypto ipsec sa map** <*name*>
**show crypto ipsec transform-set**
**show crypto ipsec transform-set** <*name*>

## Syntax Description

| | |
|---|---|
| **sa** | Displays all IPSec security associations. |
| **sa address** <*ip address*> | Optional. Displays all IPSec security associations associated with the designated peer IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| **sa map** <*name*> | Optional. Displays all IPSec security associations associated with the designated crypto map name. |
| **transform-set** | Displays all defined transform sets. |
| <*name*> | Optional. Displays information for a specific transform set. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 4.1 | Command was introduced. |

# show crypto map

Use the **show crypto map** command to display information regarding crypto map settings. Variations of this command include the following:

**show crypto map**
**show crypto map interface** <*interface*>
**show crypto map** <*name*>
**show crypto map** <*name*> <*number*>

## Syntax Description

| | |
|---|---|
| **interface** <*interface*> | Optional. Displays the crypto map settings for the specified interface. Specify an interface in the format *<interface type [slot/port \| slot/port.sub-interface id \| interface id \| interface id.sub-interface id]>*. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **show crypto map interface ?** for a complete list of valid interfaces. |
| <*name*> | Optional. Specifies a specific crypto map name. |
| <*number*> | Optional. Specifies a specific crypto map number. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 4.1 | Command was introduced. |

## Usage Examples

The following is a sample from the **show crypto map** command:

```
> enable
#show crypto map testMap
Crypto Map “testMap” 10 ipsec-ike
 Extended IP access list NewList
 Peers:
   63.97.45.57
 Transform sets:
   esp-des
 Security-association lifetimes:
   0 kilobytes
   86400 seconds
 No PFS group configured
 Interfaces using crypto map testMap:
   eth 0/1
```

# show debugging

Use the **show debugging** command to display a list of all activated debug message categories.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following is a sample output from the **show debugging** command:

```
>enable
#show debugging
debug access-list MatchAll
debug firewall
debug ip rip
debug frame-relay events
debug frame-relay llc2
debug frame-relay lmi
```

# show demand

Use the **show demand** command to display information regarding demand routing parameters and statistics. Variations of this command include the following:

**show demand**
**show demand interface demand** <*interface*>
**show demand resource pool**
**show demand resource pool** <*name*>
**show demand sessions**

## Syntax Description

| | |
|---|---|
| **interface** <*interface*> | Optional. Displays information for a specific demand routing interface. Valid range is 1 to 1024. Type **show demand interface ?** for a list of valid interfaces. |
| **resource pool** | Displays all resource pool information. |
| <*name*> | Optional. Displays resource pool information for a specific resource pool name. |
| **sessions** | Displays active demand sessions. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following is example output from the **show demand interface** command:

```
>enable
#show demand int 1
Demand 1 is UP (connected)
 Configuration:
   Keep-alive is set (10 sec.)
   Admin MTU = 1500
   Mode: Either, 1 dial entries, idleTime = 120, fastIdle = 20
   Resource pool demand
   No authentication configured
   IP address 10.100.0.2 255.255.255.0
 Connect Sequence: Successes = 0, Failures = 0
   Seq   DialString   Technology Successes Busys NoAnswers NoAuths InUse
    5     5552222        ISDN      0     0        0       0
 Current values:
   Local IP address 10.100.0.2, Peer IP address 10.100.0.1
```

```
   Seconds until disconnect: 63
   Queueing method: weighted fair
   Output queue: 0/1/428/64/0 (size/highest/max total/threshold/drops)
     Conversations 0/1/256 (active/max active/max total)
     Available Bandwidth 48 kilobits/sec
   Bandwidth=64 Kbps
 Link through bri 1/3, Uptime 0:01:10
   IN: Octets 588, Frames 19, Errors 0
   OUT: Octets 498, Frames 18, Errors 0
   Last callerID 2565552222, last called num 5552222
```

The following is example output from the **show demand interface demand** command:

```
>enable
#show demand interface demand 1
demand 1
Idle timer (120 secs), Fast idle timer (20 secs)

Dialer state is data link layer up
Dial reason: answered

Interface bound to resource bri 1/3
Time until disconnect 105 secs
Current call connected 00:00:27
Connected to 2565552222

Number of active calls = 1
Interesting Traffic = list junk

 Connect Sequence: Successes = 0, Failures = 0
  Seq     DialString      Technology    Successes     Busys  NoAnswers    NoAuths    InUse
   5       5552222           ISDN            0           0          0          0
```

The following is example output from the **show demand resource pool** command:

```
>enable
#show demand resource pool
Pool demand
        Resources:          bri 1/3, bri 2/3
        Demand Interfaces:  demand 1
```

The following is example output from the **show demand sessions** command:

```
>enable
#show demand sessions
Session 1
Interface demand 1
Local IP address = 10.100.0.2
Remote IP address = 10.100.0.1
Remote Username =
Dial reason: ip (s=, d=)
Link 1
 Dialed number = 5552222
 Resource interface = bri 1/3, Multilink not negotiated
 Connect time: 0:0:13
 Idle Timer: 119
```

# show dial-backup interfaces

Use the **show dial-backup interfaces** command to display all configured dial-backup interfaces and the associated parameters for each.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include PPP dial backup. |

## Usage Examples

The following example enters the Enable mode and uses the show command to display dial-backup interface information:

```
>enable
#show dial-backup interfaces
Dial-backup interfaces...
fr 1.16 backup interface:
 Backup state: idle
 Backup protocol: PPP
Call mode: originate
 Auto-backup: enabled
 Auto-restore: enabled
 Priority: 50
 Backup delay: 10 seconds
 Restore delay: 10 seconds
 Connect timeout: 60 seconds
Redial retries:   unlimited
 Redial delay:    10 seconds
Backup enabled all day on the following days:
    Sunday Monday Tuesday Wednesday Thursday Friday Saturday
Backup phone number list:
   Number      Call Type     min/max DS0s     Backup I/F
   5551212     analog        1/1              ppp 2
```

# show dialin interfaces

Use the **show dialin interfaces** command to display information regarding remote console dialin.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 4.1 | Command was introduced. |

## Usage Examples

The following is sample output from the **show dialin interfaces** command:

```
>enable
#show dialin interfaces
Dialin interfaces...
modem 1/3 dialin interface:
 Connection Status: Connected
Caller ID info: name-John Smith number-5551212 time-14:23:10 2/17/2003
```

# show dynamic-dns

Use the **show dynamic-dns** command to show information related to the dynamic domain naming system (DNS) configuration.

## Syntax Description

No subcommands.

## Default Values

No default is necessary for this command.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Usage Examples

The following is sample output from this command:

```
>enable
#show dynamic-dns
eth 0/1:
 Hostname: host
 Is Updated: no
 Last Registered IP: 10.15.221.33
 Last Update Time: 00:00:00 UTC Thu Jan 01 1970
```

# show event-history

Use the **show event-history** command to display all entries in the current local event-history log.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The event history provides useful information regarding the status of the system and individual port states. Use the event history as a troubleshooting tool when identifying system issues. The following is a sample event-history log.

```
>enable
#show event-history
Using 526 bytes
2002.07.12 15:34:01 T1.t1 1/1 Yellow
2002.07.12 15:34:01 INTERFACE_STATUS.t1 1/1 changed state to down.
2002.07.12 15:34:02 T1.t1 1/1 No Alarms
2002.07.12 15:34:02 INTERFACE_STATUS.t1 1/1 changed state to up.
2002.07.12 15:34:03 INTERFACE_STATUS.eth 0/1 changed state to up.
2002.07.12 15:34:10 OPERATING_SYSTEM Warm Start
2002.07.12 15:34:12 PPP.NEGOTIATION LCP up
2002.07.12 15:34:12 PPP.NEGOTIATION IPCP up
```

# show fan-tach

Use the **show fan-tach** command to view the unit's current fan speed.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Usage Examples

The following example shows the current fan speed:

```
>enable
#show fan-tach
```

| Fan Tach (in rpm) | Current | Min | Max | Avg |
|---|---|---|---|---|
| Processor | 8160 | 8100 | 17804 | 8544 |
| Chassis 1 | 3060 | 3060 | 31380 | 4237 |
| Chassis 2 | 3120 | 3060 | 31560 | 4277 |

# show file

Use the **show file** command to display a specified file (located in either compact flash or flash memory) to the terminal screen. Variations of this command include:

**show file** *<filename>*
**show file** *<filename>* **checksum**
**show file cflash** *<filename>*
**show file cflash** *<filename>* **checksum**
**show file flash** *<filename>*
**show file flash** *<filename>* **checksum**


*To display files located in the flash memory on products with compact flash capability, the **flash** keyword must be specified whether or not a compact flash card is installed.*

## Syntax Description

| | |
|---|---|
| *<filename>* | Displays information on the specified file. Wildcard entries (such as ***.biz**) are not valid for the **show file** command. |
| **cflash** | Specifies a file located in compact flash memory. |
| **flash** | Specifies a file located in flash memory. |
| **checksum** | Optional. Displays the Message Digest 5 (MD5) checksum of the specified file. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command introduced. |
| Release 12.1 | Command expanded to include the cflash option. |

## Usage Examples

The following is a sample **show file cflash** output:

>**enable**
**#show file cflash startup-config**
Router#show file startup-config
Using 2558 bytes
!
!
hostname “Router”
enable password password
!
clock timezone -6-Central-Time
!
ip subnet-zero
ip classless
ip routing
!
no auto-config
!
event-history on
no logging forwarding
no logging email
logging email priority-level info
!
no service password-encryption
!
username “admin” password “password”
!
--MORE

# show flash

Use the **show flash** command to display a list of all files currently stored in flash memory. Variations of this command include:

**show flash**
**show flash** *<filename>*

## Syntax Description

| | |
|---|---|
| *<filename>* | Optional. Displays details for a specified file located in flash memory. Enter a wildcard (such as ***.biz**) to display the details for all files matching the entered pattern. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following is a sample **show flash** output:

```
>enable
#show flash
Files:
 245669 010100boot.biz
1141553 new.biz
    821 startup-config
   1638 startup-config.old
1175679 020016.biz
    821 startup-config.bak
2572304 bytes used 4129776 available 6702080 total
```

# show frame-relay fragment

Use the **show frame-relay fragment** command to display detailed fragmentation statistics for Frame Relay sub-interfaces with FRF.12 fragmentation enabled. Variations of this command include:

**show frame-relay fragment**
**show frame-relay fragment interface frame-relay** <*sub-interface*>

## Syntax Description

| | |
|---|---|
| **interface frame-relay** <*sub-interface*> | Optional. Displays detailed fragmentation statistics for the specified Frame Relay sub-interface. Sub-interfaces are expressed in the format *interface id.sub-interface id*. For example, **fr 1.16**. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Usage Examples

The following are sample outputs from various **show frame-relay fragment** commands:

```
>enable
#show frame-relay fragment
interface   dlci   frag_size   rx_frag   tx_frag   dropped_frag
fr 1.1      17     100         46        48        0
fr 1.2      18     200         42        21        0
```

```
>enable
#show frame-relay fragment frame-relay 1.1
DLCI =  17 FRAGMENT SIZE = 100
rx frag. pkts                    46    tx frag. pkts                48
rx frag. bytes                 4598    tx frag. bytes             4724
rx non-frag. pkts                18    tx non-frag. pkts            28
rx non-frag. bytes             1228    tx non-frag. bytes         1960
rx assembled pkts                23    tx pre-fragment pkts         34
rx assembled bytes             5478    tx pre-fragment bytes      6324
dropped reassembling pkts         0    dropped fragmenting pkts      0
rx out-of-sequence fragments      0
rx unexpected beginning fragment  0
```

# show frame-relay

Use the **show frame-relay** command to display configuration and status parameters for configured virtual Frame Relay interfaces. Variations of this command include the following:

**show frame-relay lmi**
**show frame-relay pvc**
**show frame-relay pvc interface frame-relay** *<interface>*
**show frame-relay pvc interface frame-relay** *<interface>* **realtime**
**show frame-relay pvc realtime**


*Using the **realtime** argument for this command can adversely affect the performance or your unit.*

## Syntax Description

| | |
|---|---|
| **lmi** | Displays Link Management Interface (LMI) statistics for each virtual Frame Relay interface. |
| **pvc** | Displays Permanent Virtual Circuit (PVC) configuration and statistics for all virtual Frame Relay interfaces (or a specified interface). |
| **interface frame-relay** | Displays Frame Relay PVC statistics for a specific Frame Relay interface. |
| *<interface>* | Specifies the virtual Frame Relay interface (for example, **fr 1**). |
| **realtime** | Optional. Displays full-screen output in real time. Refer to the *Functional Notes* below for more information. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 10.1 | **Realtime** option was introduced. |

## Functional Notes

Use the **realtime** argument for this command to display full-screen output in real time. Information is continuously updated on the console until you either freeze the data (by pressing the **F** key) or exit **realtime** mode (by pressing **Ctrl-C)**. If there is not enough room on the screen for all available data, the information will truncate at the bottom of the screen. In order to maximize the amount of data displayed, increase the terminal length (using the **terminal length** command; refer to *terminal length <number>* on page 410).

## Usage Examples

The following are sample outputs from various **show frame-relay** commands:

```
>enable
#show frame-relay lmi
LMI statistics for interface FR 1 LMI TYPE = ANSI
Num Status Enq. Sent 79      Num Status Msgs Rcvd 71
Num Update Status Rcvd 12    Num Status Timeouts 5

>enable
#show frame-relay pvc
Frame Relay Virtual Circuit Statistics for interface FR 1
             Active     Inactive   Deleted    Static
local        2          0          0          2
DLCI = 16 DLCI USAGE = LOCAL PVC STATUS = ACTIVE INTERFACE = FR 1.16
MTU: 1500
input pkts: 355                  output pkts: 529                in bytes: 23013
out bytes: 115399                dropped pkts: 13                in FECN pkts: 0
in BECN pkts: 0                  in DE pkts: 0                   out DE pkts: 0
pvc create time: 00:00:00:12     last time pvc status changed: 00:00:13:18
DLCI = 20 DLCI USAGE = LOCAL PVC STATUS = ACTIVE INTERFACE = FR 1.20
MTU: 1500
input pkts: 0                    output pkts: 44                 in bytes: 0
out bytes: 22384                 dropped pkts: 11                in FECN pkts: 0
in BECN pkts: 0                  in DE pkts: 0                   out DE pkts: 0
pvc create time: 00:00:01:25     last time pvc status changed: 00:00:13:18
```

# show frame-relay multilink

Use the **show frame-relay multilink** command to display information associated with the Frame Relay multilink interface. Variations of this command include:

**show frame-relay multilink**
**show frame-relay multilink detailed**
**show frame-relay multilink** <*interface*>
**show frame-relay multilink** <*interface*> **detailed**
**show frame-relay multilink interface frame-relay** <*sub-interface*>

## Syntax Description

| | |
|---|---|
| <*interface*> | Optional. Specifies the display of information for a specific interface. Enter the **show frame-relay multilink ?** command for a complete list of interfaces. |
| **detailed** | Optional. Displays more detailed information. |
| **interface frame-relay** <*sub-interface*> | Optional. Displays detailed fragmentation statistics for the specified Frame Relay sub-interface. Sub-interfaces are expressed in the format *interface id.sub-interface id*. For example, **fr 1.16**. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Usage Examples

The following is a sample output from this command:

**>enable**
**#show frame-relay multilink**
Bundle: frame-relay 1 is DOWN; class A bundle
Near-end BID: MFR1; Far-end BID: unknown

# show garp timer

Use the **show garp timer** command to see the current configured Generic Attribute Registration Protocol (GARP) application timer values.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Usage Examples

The following example displays the current configured GARP application timer values:

```
>enable
#show garp timer
Timer          Timer Value (milliseconds)
--------       --------------------------
Join            200
Leave           600
LeaveAll        10000
```

# show gvrp configuration

Use the **show gvrp configuration** command to show a GARP VLAN Registration Protocol (GVRP) configuration summary for the switch.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Usage Examples

The following example displays a GVRP configuration summary for the switch:

```
>enable
#show gvrp configuration
Global GVRP Configuration:
GVRP Feature is currently enabled globally.
GVRP Timers (milliseconds)
Join 200
Leave 600
LeaveAll 20000
Port based GVRP Configuration:
GVRP enabled ports
---------------------------------------------------
eth 0/24

#
```

# show gvrp statistics

Use the **show gvrp statistics** command to show statistics related to GARP VLAN Registration Protocol (GVRP). Variations of this command include:

**show gvrp statistics**
**show gvrp statistics interface** <*interface*>

## Syntax Description

| | |
|---|---|
| **interface** <*interface*> | Optional. Shows the information for the specified interface. Specify an interface in the format <*interface type [slot/port* \| *slot/port.sub-interface id* \| *interface id* \| *interface id.sub-interface id]*>. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **show gvrp statistics interface ?** for a complete list of applicable interfaces. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Usage Examples

The following example displays statistics related to GVRP for Ethernet interface 0/24:

```
>enable
#show gvrp statistics interface ethernet 0/24
Name: eth 0/24
Join Empty Received: 0
Join In Received: 272
Empty Received: 30
Leave Empty Received: 0
Leave In Received: 0
Leave All Received: 28
Join Empty Transmitted: 0
Join In Transmitted: 286
Empty Transmitted: 28
Leave Empty Transmitted: 0
Leave In Transmitted: 0
Leave All Transmitted: 115
#
```

# show hosts

Use the **show hosts** command to display information such as the domain name, name lookup service, a list of name server hosts, and the cached list of host names and addresses on the network to which you can connect. Variations of this command include:

**show hosts**
**show hosts verbose**

## Syntax Description

| | |
|---|---|
| **verbose** | Optional. Enables detailed messaging. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Functional Notes

The list below describes the fields contained in the host table:

- Flags: Indicate whether the entry is permanent (P) or temporary (T) and if the entry is OK or expired (EXP).
- Age: Indicates the age of the entry.
- Type: Shows the protocol type.
- Address: Displays the IP address for the entry.

## Usage Examples

The following example is sample output from the **show hosts** command:

>**enable**
**#show hosts**
Name/address lookup uses domain name service
DNS Proxy is disabled
Default domain is not set
Name servers are 1.1.1.1 2.2.2.2

| Host | Flags | Age | Type | Address |
|---|---|---|---|---|
| Example1 | (P OK) | - - | IP | 1.1.1.1 |
| Example2 | (P OK) | - - | IP | 2.2.2.2 |

# show interfaces

Use the **show interfaces** command to display configuration parameters and current statistics for all switch port interfaces (or a specified switch port interface). These commands are valid only on switch ports. Variations of this command include the following:

**show interfaces description**
**show interfaces status**
**show interfaces** <*interface*> **switchport**
**show interfaces** <*interface*> **switchport vlans**

## Syntax Description

| | |
|---|---|
| **description** | Displays information such as description, administrative status, line protocol status, and description for all the interfaces. |
| **status** | Displays information such as description, type, status, VLAN, speed, and duplex for all the Ethernet interfaces only. |
| **switchport** | Displays information such as description, administrative status, line protocol status, and description for all the switch ports. |
| <*interface*> | Optional. Specifies a switch port interface in the format <*interface type [slot/port* \| *slot/port.sub-interface id* \| *interface id* \| *interface id.sub-interface id]*>. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **show switchport ?** for a complete list of valid interfaces. |
| **vlans** | Optional. Displays the VLAN membership information for a specific switch port or series of switch ports. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 6.1 | Command expanded to include the switchport option. |
| Release 10.1 | Command was expanded to include the **vlans** option. |
| Release 11.1 | Description and status options were introduced. |

## Usage Examples

The following is sample output from the **show interfaces description** command:

```
>enable
#show interfaces description
```

| Interface | Status | Protocol | Description |
|---|---|---|---|
| eth 0/1 | Admin Up | Line Up | Desk 1 |
| eth 0/2 | Admin Up | Line Up | Desk 2 |
| eth 0/3 | Admin Up | Line Up | Desk 3 |
| eth 0/4 | Admin Up | Line Up | Desk 4 |
| eth 0/5 | Admin Up | Line Up | Desk 5 |
| eth 0/6 | Admin Up | Line Up | Desk 6 |
| eth 0/7 | Admin Up | Line Up | Desk 7 |
| eth 0/8 | Admin Up | Line Down | Desk 8 |
| eth 0/9 | Admin Up | Line Up | Desk 9 |
| eth 0/10 | Admin Up | Line Up | Desk 10 |
| eth 0/11 | Admin Up | Line Up | Desk 11 |
| eth 0/12 | Admin Up | Line Up | Desk 12 |
| eth 0/13 | Admin Up | Line Up | Desk 13 |
| eth 0/14 | Admin Up | Line Up | Desk 14 |
| eth 0/15 | Admin Up | Line Up | Desk 15 |
| eth 0/16 | Admin Up | Line Up | Desk 16 |
| eth 0/17 | Admin Up | Line Up | Desk 17 |
| eth 0/18 | Admin Up | Line Up | Desk 18 |
| eth 0/19 | Admin Up | Line Up | Desk 19 |
| eth 0/20 | Admin Up | Line Up | Desk 20 |
| eth 0/21 | Admin Up | Line Up | Desk 21 |
| eth 0/22 | Admin Up | Line Up | Desk 22 |
| eth 0/23 | Admin Up | Line Up | Desk 23 |
| eth 0/24 | Admin Up | Line Up | Desk 24 |
| giga-eth 0/1 | Admin Up | Line Up | Uplink Trunk |
| giga-eth 0/2 | Admin Up | Line Down | Unused |

The following is sample output from the **show interfaces switchport** command:

>**enable**
**#show interfaces switchport**
Name: eth 0/1
Switchport: enabled
Administrative mode: access
Negotiation of Trunking: access
Access mode VLAN: 1
Trunking Native mode VLAN: 1
Trunking VLAN Enabled: 1-4094
Name: eth 0/2
Switchport: enabled
Administrative mode: access
Negotiation of Trunking: access
Access mode VLAN: 12.....

# show interfaces *<interface>*

Use the **show interfaces** command to display configuration parameters and current statistics for all interfaces (or a specified interface). Variations of this command include the following:

**show interfaces** *<interface>*
**show interfaces** *<interface>* **performance-statistics**
**show interfaces** *<interface>* **performance-statistics** *<x-y>*
**show interfaces** *<interface>* **performance-statistics total-24-hour**
**show interfaces** *<interface>* **realtime**
**show interfaces** *<interface>* **verbose**
**show interfaces** *<interface>* **version**

*Not all subcommands apply to all interfaces. Type **show interfaces <interface> ?** for a list of valid subcommands for the specified interface.*

## Syntax Description

| | |
|---|---|
| *<interface>* | Specifies an interface in the format *<interface type [slot/port \| slot/port.sub-interface id \| interface id \| interface id.sub-interface id]>*. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **show interfaces ?** for a complete list of valid interfaces. |
| **performance-statistics** | Optional. Displays all 96 stored intervals. |
| **performance-statistics** *<x-y>* | Shows a specified interval (x) or range of intervals (x-y). |
| **performance-statistics total-24-hour** | Optional. Displays the current 24-hour totals. |
| **realtime** | Displays full-screen output in real time. Refer to the *Functional Notes* below for more information. |
| **verbose** | Displays detailed configuration information on the terminal screen (versus only the non-default values). |
| **version** | Optional. Displays current version information (e.g., model and list number, software version, etc.) for the interface. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 6.1 | Command was updated to include performance-statistics option. |
| Release 9.1 | Command was expanded to include HDLC and tunnel interfaces. |
| Release 10.1 | The realtime option and PRI interface were added. |
| Release 11.1 | Description, status, and verbose options were introduced. The demand, FXO, and serial interfaces were added. |

## Functional Notes

Use the **realtime** argument for this command to display full-screen output in real time. Information is continuously updated on the console until you either freeze the data (by pressing the **F** key) or exit **realtime** mode (by pressing **Ctrl-C**). If there is not enough room on the screen for all available data, the information will truncate at the bottom of the screen. In order to maximize the amount of data displayed, increase the terminal length (using the **terminal length** command; refer to *terminal length <number>* on page 410).

## Usage Examples

The following are samples from various **show interfaces** commands:

```
>enable
#show interfaces t1 1/1
t1 1/1 is UP
 T1 coding is B8ZS framing is ESF
 Clock source is line FDL type is ANSI
 Line build-out is 0dB
 No remote loopbacks No network loopbacks
DS0 Status: 12345678901234567890123
            NNNNNNNNNNNNNNNNNNNNNNN
Line Status: -- No Alarms --
Current Performance Statistics:
   0 Errored Seconds 0 Bursty Errored Seconds
   0 Severely Errored Seconds 0 Severely Errored Frame Seconds
   0 Unavailable Seconds 0 Path Code Violations
   0 Line Code Violations 0 Controlled Slip Seconds
   0 Line Errored Seconds 0 Degraded Minutes

#show interfaces modem 1/2
modem 1/2 is UP
 Line status: on-hook
 Caller ID will be used to route incoming calls
   0 packets input 0 bytes 0 no buffer
   0 runts 0 giants 0 throttles
   0 input errors 0 CRC 0 frame
   0 abort 0 ignored 0 overruns
```

```
   0 packets output 0 bytes 0 underruns
   0 input clock glitches 0 output clock glitches
0 carrier lost 0 cts lost
```

**#show interfaces eth 0/1**

```
Ip address is 10.200.1.50
 Netmask is 255.255.0.0
 MTU is 1500
 Fastcaching is Enabled
 RIP Authentication is Disabled
 RIP Tx uses global version value
 RIP Rx uses global version value
```

**#show interfaces dds 1/1**

```
dds 1/1 is UP line protocol is UP
 Encapsulation FRAME-RELAY (fr 1)
 Loop rate is set to 56000 actual rate is 56000
 Clock source is line
 Data scrambling is disabled
 No Loopbacks
   75 packets input 6108 bytes 0 no buffer
   0 runts 0 giants 0 throttles
   0 input errors 0 CRC 0 frame
   0 abort 0 ignored 0 overruns
   81 packets output 11496 bytes 0 underruns
   0 input clock glitches 0 output clock glitches
   0 carrier lost 0 cts lost
```

**#show interfaces fr 1**

```
TDM group 10 line protocol is UP
Encapsulation FRAME-RELAY (fr 1)
   463 packets input 25488 bytes 0 no buffer
   0 runts 0 giants 0 throttles
   0 input errors 0 CRC 0 frame
   0 abort 0 ignored 0 overruns
   864 packets output 239993 bytes 0 underruns
   0 input clock glitches 0 output clock glitches
   0 carrier lost 0 cts lost
Line Status: -- No Alarms --
Current Performance Statistics:
   0 Errored Seconds 0 Bursty Errored Seconds
   0 Severely Errored Seconds 0 Severely Errored Frame Seconds
   0 Unavailable Seconds 0 Path Code Violations
   0 Line Code Violations 0 Controlled Slip Seconds
   0 Line Errored Seconds 0 Degraded Minutes
```

**#show interfaces fr 1.100**
fr 1.100 is Active
Ip address is 63.97.45.57, mask is 255.255.255.248
Interface-dlci is 100
MTU is 1500 bytes, BW is 96000 Kbit (limited)
Average utilization is 53%

**#show interfaces shdsl 1/1**
shdsl 1/1 is UP, line protocol is DOWN
Encapsulation FRAME-RELAY IETF (fr 1)
Equipment type is cpe
Line rate is 2056kbps
No alarms.
SHDSL training complete. EOC is up.
No local loopbacks, No remote loopbacks
SNR margin is 18dB currently, 15dB minimum, 30dB maximum
Loop attenuation is 1dB currently, 1dB minimum, 1dB maximum
Current 15-minute performance statistics (115 seconds elapsed):
0 code violations, 0 loss of sync word seconds
0 errored seconds, 0 severely errored seconds
0 unavailable seconds
Packet Statistics:
0 packets input, 0 bytes, 0 no buffer
0 runts, 0 giants, 0 throttles
0 input errors, 0 CRC, 0 frame
0 abort, 0 ignored, 0 overruns
32 packets output, 0 bytes, 0 underruns
0 input clock glitches, 0 output clock glitches

*Note: If the user has configured a **Bc** and **Be** value on the virtual circuit, the bandwidth (**BW**) displayed is the sum of those values (Bc + Be). If not, the value for **BW** is the speed of the interface. The **Average utilization** displayed is the average utilization of the displayed bandwidth. If the bandwidth number is the Bc + Be value, the (**limited**) text appears (as shown above).

# show interfaces adsl <*slot/port*>

Use the **show interfaces adsl** command to display information related to the ADSL port. Variations of this command include:

**show interfaces adsl** <*slot/port*>
**show interfaces adsl** <*slot/port*> **information**
**show interfaces adsl** <*slot/port*> **information atuc**
**show interfaces adsl** <*slot/port*> **information atur**
**show interfaces adsl** <*slot/port*> **information bit-allocation**
**show interfaces adsl** <*slot/port*> **performance-statistics**
**show interfaces adsl** <*slot/port*> **performance-statistics** <*x-y*>
**show interfaces adsl** <*slot/port*> **performance-statistics total-24-hour**
**show interfaces adsl** <*slot/port*> **performance-statistics total-previous-24-hour**
**show interfaces adsl** <*slot/port*> **version**

## Syntax Description

| | |
|---|---|
| <*slot/port*> | Specifies ADSL interface slot and port number. |
| **information atuc** | Shows ADSL interface remote information. |
| **information atur** | Shows ADSL local information. |
| **information bit-allocation** | Shows ADSL DMT bit-allocation table. |
| **performance-statistics** | Optional. Displays all 96 stored intervals. |
| **performance-statistics** <*x-y*> | Shows a specified interval (x) or range of intervals (x-y). |
| **performance-statistics total-24-hour** | Optional. Displays the current 24-hour totals. |
| **version** | Optional. Displays current version information (e.g., model and list number, software version, etc.) for the interface. |

## Default Values

No default is necessary for this command.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Usage Examples

The following example shows sample output for this command:

```
>enable
#show interfaces adsl 1/1 information
adsl 1/1 line information
adsl 1/1 Local Line Information
 Vendor Id:                    00000000
 Serial Number:                00000000
 Firmware Version:
 ADSL Capabilities             G.DMT, G.LITE, ADSL2, ADSL2+
adsl 0/1 Remote Line Information
 Vendor Id:                    00000000
 Serial Number:                00000000
 Firmware Version:             0
 ADSL Capabilities             G.DMT, G.LITE, ADSL2, ADSL2+
```

# show ip access-lists

Use the **show ip access-lists** command to display all configured IP access control lists in the system. Variations of this command include:

**show ip access-lists**
**show ip access-lists** *<name>*

## Syntax Description

| | |
|---|---|
| *<name>* | Optional. Specifies a particular access control list to display. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |

## Functional Notes

The **show ip access-lists** command displays all configured IP access control lists in the system. All entries in the access control list are displayed, and a counter indicating the number of packets matching the entry is listed.

## Usage Examples

The following is a sample output from the **show ip access-lists** command:

```
>enable
#show ip access-lists
Standard IP access list MatchAll
   permit host 10.3.50.6 (0 matches)
   permit 10.200.5.0 wildcard bits 0.0.0.255 (0 matches)
Extended IP access list UnTrusted
   deny   icmp 10.5.60.0 wildcard bits 0.0.0.255 any source-quench (0 matches)
   deny   tcp any any   (0 matches)
```

# show ip arp

Use the **show ip arp** command to display the Address Resolution Protocol (ARP) table. Variations of this command include:

**show ip arp**
**show ip arp realtime**

> NOTE *Using the **realtime** argument for this command can adversely affect the performance or your unit.*

## Syntax Description

| | |
|---|---|
| **realtime** | Optional. Displays full-screen output in real time. Refer to the *Functional Notes* below for more information. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |
| Release 10.1 | The real time display option was introduced. |

## Functional Notes

Use the **realtime** argument for this command to display full-screen output in real time. Information is continuously updated on the console until you either freeze the data (by pressing the **F** key) or exit **realtime** mode (by pressing **Ctrl-C**). If there is not enough room on the screen for all available data, the information will truncate at the bottom of the screen. In order to maximize the amount of data displayed, increase the terminal length (using the **terminal length** command; refer to *terminal length <number>* on page 410).

## Usage Examples

The following is a sample output of the **show ip arp** command:

**>enable**
**#show ip arp**

| ADDRESS | TTL (min) | MAC ADDRESS | LAST UPDATED (min) |
|---|---|---|---|
| 192.168.30.36 | 13 | 00:E0:7D:88:1A:B9 | 4260 |
| 192.168.30.253 | 17 | 02:60:8C:DD:0A:CE | 4264 |
| 224.0.0.9 | 71578541 | 01:00:5E:00:00:09 | 0 |

# show ip as-path-list

Use the **show ip as-path-list** command to display any AS path lists that have been configured in the router, along with any permit and deny clauses in each list. Variations of this command include:

**show ip as-path-list**
**show ip as-path-list** <*name*>

## Syntax Description

| | |
|---|---|
| <*name*> | Optional. Specifies that the command display only the list matching the specified AS path listname. If not specified, all AS path lists are displayed. |

## Default Values

By default, this command displays all AS path lists.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

In the following example, all AS path lists defined in the router are displayed.

```
>enable
#show ip as-path-list
ip as-path-list AsPathList1:
 permit 100
 permit 200
 permit 300
 deny 6500
ip as-path-list AsPathList2:
 permit 400
 permit 500
```

In the following example, only the AS Path List with the name **AsPathList2** is displayed.

```
>enable
#show ip as-path-list AsPathList2
ip as-path-list AsPathList2:
 permit 400
 permit 500
```

# show ip bgp

Use the **show ip bgp** command to display details about the specified route, including the advertising router IP address, router ID, and the list of neighbors to which this route is being advertised. Variations of this command include:

**show ip bgp**
**show ip bgp** <*ip address*>
**show ip bgp** <*ip address*> <*subnet mask*>
**show ip bgp summary**

## Syntax Description

| | |
|---|---|
| <*ip address*> | Specifies a valid IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| <*subnet mask*> | Optional. Specifies the subnet mask that corresponds to a range of IP addresses (network) or a specific host. Subnet masks can be expressed in dotted decimal notation (for example, **255.255.255.0**) or as a prefix length (for example, **/24**). |
| **summary** | Optional. Displays a summary of the BGP route table. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Usage Examples

The following example shows detailed output of this command:

```
>enable
#show ip bgp 10.15.240.0/28
BGP routing table entry for 10.15.240.0/28
Paths: (1 available, best #1)
 Advertised to peers:
 1.1.5.10
 100 1
   10.15.43.17 from 10.15.43.17 (8.1.1.1)
     Origin IGP, metric 2, valid, external, best
```

The following sample output of the **show ip bgp summary** command shows a summarized list of the configured BGP neighbors as well as their status and statistics.

>**enable**
**#show ip bgp summary**
BGP router identifier 192.168.3.1, local AS number 304
8 network entries, 5 paths, and 23 BGP path attribute entries

| Neighbor | V | AS | MsgRcvd | MsgSent | InQ | OutQ | Up/Down | State/PfxRcd |
|---|---|---|---|---|---|---|---|---|
| 10.22.131.1 | 4 | 302 | 95 | 104 | 0 | 0 | 01:30:06 | 9 |
| 10.22.131.9 | 4 | 302 | 97 | 105 | 0 | 0 | 01:30:07 | 21 |
| 10.22.132.9 | 4 | 303 | 200 | 179 | 0 | 0 | 02:43:09 | 21 |
| 10.22.134.1 | 4 | 304 | 166 | 178 | 0 | 0 | 02:43:15 | 3 |
| 10.22.134.10 | 4 | 304 | 174 | 179 | 0 | 0 | 02:43:24 | 7 |
| 10.22.134.26 | 4 | 304 | 172 | 174 | 0 | 0 | 02:41:43 | 10 |
| 10.22.134.34 | 4 | 304 | 164 | 174 | 0 | 0 | 02:41:40 | 4 |

# show ip bgp community

Use the **show ip bgp community** command to display only those routes learned via Border Gateway Protocol (BGP) that match the community numbers specified in the command. If no communities are specified, all BGP routes are shown. Variations of this command include:

**show ip bgp community**
**show ip bgp community** <*number*>
**show ip bgp community** <*number*> **exact**
**show ip bgp community internet**
**show ip bgp community internet exact**
**show ip bgp community local-as**
**show ip bgp community local-as exact**
**show ip bgp community no-advertise**
**show ip bgp community no-advertise exact**
**show ip bgp community no-export**
**show ip bgp community no-export exact**

## Syntax Description

| | |
|---|---|
| <*number*> | Optional. Displays routes that contain this value in their community attribute. This is a numeric value that can be an integer from 1 to 4,294,967,295 or string in the form “aa:nn”, where the value of “aa” is the AS number and the value of “nn” is the community number. Multiple community-number parameters can be present in the command. |
| **exact** | Optional. Displays BGP routes with the community numbers specified and *only* those specified. |
| **internet** | Optional. Displays routes that contain this value in their community attribute. This represents the well-known reserved community number for the INTERNET community. |
| **local-as** | Optional. Displays routes that contain this value in their community attribute. This represents the well-known reserved community number for NO_EXPORT_SUBCONFED. Routes containing this attribute should not be advertised to external BGP peers. |
| **no-advertise** | Optional. Displays routes containing this value in the community attribute. This represents the well-known reserved community number for NO_ADVERTISE. Routes containing this attribute should not be advertised to any BGP peer. |
| **no-export** | Optional. Displays routes containing this value in the community attribute. This represents the well-known reserved community number for NO_EXPORT. Routes containing this attribute should not be advertised to BGP peers outside a confederation boundary. |

## Default Values

By default, this command displays all BGP routes.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

In the following example, all BGP routes are displayed whose community numbers match those listed in the **show ip bgp community** command.

```
>enable
#show ip bgp community local-as 10:405
BGP local router ID is 10.22.131.241, local AS is 302.
Status codes: * valid, > best, i - internal, o - local
Origin codes: i - IGP, e - EGP, ? - incomplete
   Network          Next Hop         Metric     LocPrf     Path
   10.22.152.20/30  10.22.131.10     304                   302 300 1 3 4 i
   10.22.152.24/29  10.22.131.10     304                   302 300 1 3 4 5 i
   10.22.152.36/30  10.22.131.10     304                   302 300 1 3 4 i
   10.22.152.52/30  10.22.131.10     304                   302 300 1 3 4 i
   11.0.0.0/30      10.22.131.10     304                   302 300 1 3 4 6 i
   12.0.0.0/30      10.22.131.10     304                   302 300 1 3 4 6 i
   13.0.0.0/30      10.22.131.10     304                   302 300 1 3 4 6 i
   14.0.0.0/30      10.22.131.10     304                   302 300 1 3 4 6 i
Total RIB entries = 8
```

Information displayed includes: the ID of this router and its Autonomous System (AS) number; the destination Network address of the route learned; the Next Hop address to that network; the Metric; the Local Preference value (set using the **set local-preference** command); and the AS Path to the destination network.

The following is a sample output for the **show-ip bgp community** command with an exact match specified: BGP routes with the community numbers specified and *only* those specified are shown

```
>enable
#show ip bgp community 1001 2001 3001 exact
BGP local router ID is 192.168.9.1, local AS is 252.
Status codes: * valid, > best, i - internal, o - local
Origin codes: i - IGP, e - EGP, ? - incomplete
   Network          NextHop          Metric     LocPrf     Path
   * 192.168.11.0/24  10.22.27.251                        249 251 i
   * 192.168.12.0/24  10.22.27.251                        249 251 i
   *> 192.168.32.0/24 10.22.27.249                        249 i
   *> 192.168.33.0/24 10.22.27.249                        249 i
Total RIB entries = 4
```

# show ip bgp community-list

Use the **show ip bgp community-list** command to display Border Gateway Protocol (BGP) routes that are permitted by the specified community list. Variations of this command include:

**show ip bgp community-list** <*name*>
**show ip bgp community-list** <*name*> **exact**

## Syntax Description

| | |
|---|---|
| <*name*> | Specifies the name of the community list whose routes you wish to see. |
| **exact** | Optional. Restricts the routes displayed to only those whose community lists exactly match those specified in the named community list. If this parameter is omitted, all routes matching any part of the specified community list will be displayed. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Functional Notes

Information displayed includes the ID of this router and its Autonomous System (AS) number, the destination Network address of the route learned, the Next Hop address to that network, the Metric, the Local Preference (LocPrf) value (set using the **set local-preference *** command), and the AS Path to the destination network.

## Usage Examples

In the following example, all BGP routes are displayed whose community numbers match those defined in the community list named CList1.

```
>enable
#show ip bgp community-list CList1
BGP local router ID is 10.22.131.241, local AS is 302.
Status codes: * valid, > best, i - internal, o - local
Origin codes: i - IGP, e - EGP, ? - incomplete
   Network          Next Hop        Metric  LocPrf  Path
   10.22.152.20/30  10.22.131.10     304             302 300 1 3 4 i
   10.22.152.24/29  10.22.131.10     304             302 300 1 3 4 5 i
   10.22.152.36/30  10.22.131.10     304             302 300 1 3 4 i
   10.22.152.52/30  10.22.131.10     304             302 300 1 3 4 i
   11.0.0.0/30      10.22.131.10     304             302 300 1 3 4 6 i
   12.0.0.0/30      10.22.131.10     304             302 300 1 3 4 6 i
   13.0.0.0/30      10.22.131.10     304             302 300 1 3 4 6 i
   14.0.0.0/30      10.22.131.10     304             302 300 1 3 4 6 i
   20.0.0.0/30      10.22.131.10     304             302 300 1 3 4 5 i
   21.0.0.0/30      10.22.131.10     304             302 300 1 3 4 5 i
Total RIB entries = 10
```

# show ip bgp neighbors *<ip address>*

Use the **show ip bgp neighbors** command to display information for the specified Border Gateway Protocol (BGP) neighbor. Variations of this command include the following:

**show ip bgp neighbors**
**show ip bgp neighbors** *<ip address>*
**show ip bgp neighbors** *<ip address>* **advertised-routes**
**show ip bgp neighbors** *<ip address>* **received-routes**
**show ip bgp neighbors** *<ip address>* **routes**

## Syntax Description

| | |
|---|---|
| *<ip address>* | Optional. Displays information for the specified neighbor. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**).<br>If no IP address is entered, information for all neighbors is displayed. |
| **advertised-routes** | Optional. Displays all routes being advertised to the specified neighbor. Command output is the same as for **show ip bgp** except filtered to only the BGP routes being advertised to the specified neighbor. |
| **received-routes** | Optional. Displays all routes (accepted and rejected) advertised by the specified neighbor. Routes may be rejected by inbound filters such as prefix list filters. |
| **routes** | Optional. Displays all accepted received routes advertised by the specified neighbor. Routes displayed have passed inbound filtering. This command output is the same as **show ip bgp** except the output is filtered to those learned from the specified neighbor. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Functional Notes

Entries that are not filtered by prefix lists are marked with an asterisk (*) to show they are valid. Entries that are deemed the best path to advertised route are marked with a caret (>).

## Usage Examples

The following are output variations of the **show ip bgp neighbors** command:

```
>enable
#show ip bgp neighbors
BGP neighbor is 10.15.43.17, remote AS 100, external link
Configured hold time is 180, keepalive interval is 60 seconds
Default minimum time between advertisement runs is 30 seconds
Connections established 6; dropped 5
Last reset: Interface went down
 Connection ID: 15
   BGP version 4, remote router ID 8.1.1.1
   BGP state is Established, for 01:55:05
   Negotiated hold time is 180, keepalive interval is 60 seconds
   Message statistics:
    InQ depth is 0, OutQ depth is 0
Local host: 10.15.43.18, Local port: 179
     Sent                Rcvd
     Opens:1             1
     Notifications: 0    0
     Updates: 0          8
     Keepalives: 116     116
     Unknown: 0          0
     Total: 117          125
Foreign host: 10.15.43.17, foreign port: 1048
   Flags: passive open

#show ip bgp neighbors 10.15.43.34 advertised-routes
BGP local router ID is 10.0.0.1, local AS is 101.
Status codes: * valid, > best, i - internal
Origin codes: i - IGP, e - EGP, ? - incomplete

     Network        NextHop          Metric Path
*>   1.0.0.0/8      10.15.43.17      1 100 i
*>   2.0.0.0/9      10.15.43.17      1 100 i

#show ip bgp neighbors 10.15.43.17 received-routes
BGP local router ID is 10.0.0.1, local AS is 101.
Status codes: * valid, > best, i - internal
Origin codes: i - IGP, e - EGP, ? - incomplete

     Network        NextHop          Metric Path
*>   1.0.0.0/8      10.15.43.17      1 100 i
*>   2.0.0.0/9      10.15.43.17      1 100 i
```

**#show ip bgp neighbors 10.15.43.17 routes**
BGP local router ID is 10.0.0.1, local AS is 101.
Status codes: * valid, > best, i - internal
Origin codes: i - IGP, e - EGP, ? - incomplete

| | Network | NextHop | Metric Path |
|---|---|---|---|
| *> | 1.0.0.0/8 | 10.15.43.17 | 1 100 i |
| *> | 2.0.0.0/9 | 10.15.43.17 | 1 100 |

# show ip bgp regexp *<expression>*

Use the **show ip bgp regexp** command to display a summary of the BGP route table that includes routes who autonomous system (AS) path matches the specified expression.

## Syntax Description

| | |
|---|---|
| *<expression>* | Displays routes whose autonomous system (AS) path matches the regular expression specified. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Functional Notes

Entries that are not filtered by prefix lists are marked with an asterisk (*) to show they are valid. Entries that are deemed the best path to advertised route are marked with a caret (>).

## Usage Examples

The following sample output of the **show ip bgp regexp _303_** command shows all of the entries in the BGP database that contain “303” in the AS path.

```
>enable
#show ip bgp regexp _303_
BGP local router ID is 192.168.3.1, local AS is 304.
Status codes: * valid, > best, i - internal, o - local
Origin codes: i - IGP, e - EGP, ? - incomplete
Network NextHop Metric LocPrf Path
10.22.130.8/29 10.22.132.9 303 304 302 i
* i10.22.130.240/28 0.22.132.1 100 303 300 i
* 10.22.130.240/28 10.22.132.9 303 300 i
10.22.131.0/29 10.22.132.9 303 304 302 i
10.22.131.8/29 10.22.132.9 303 304 302 i
* i10.22.131.16/29 10.22.132.1 0 100 303 i
* 10.22.131.16/29 10.22.132.9 0 303 i
* i10.22.131.240/28 10.22.132.1 100 303 300 i
* 10.22.131.240/28 10.22.132.9 303 300 i
* 10.22.132.0/29 10.22.131.1 0 302 303 i
* 10.22.132.0/29 10.22.131.9 0 302 303 i
* i10.22.132.0/29 10.22.132.1 0 100 303 i
*> 10.22.132.0/29 10.22.132.9 0 303 i
* 10.22.132.8/29 10.22.131.1 0 302 303 i
* 10.22.132.8/29 10.22.131.9 0 302 303 i
```

```
* 10.22.132.8/29 10.22.132.9 0 303 i
* i10.22.132.240/28 10.22.132.1 0 100 303 i
*> 10.22.132.240/28 10.22.132.9 0 303 i
10.22.134.0/29 10.22.132.9 303 304 i
10.22.134.8/29 10.22.132.9 303 304 i
10.22.134.16/29 10.22.132.9 303 304 i
10.22.134.24/29 10.22.132.9 303 304 i
10.22.134.32/29 10.22.132.9 303 304 i
10.22.134.40/29 10.22.132.9 303 304 i
10.22.134.48/29 10.22.132.9 303 304 i
10.22.134.56/29 10.22.132.9 303 304 i
10.22.134.64/29 10.22.132.9 303 304 i
10.22.134.80/29 10.22.132.9 303 304 i
10.22.135.0/29 10.22.132.9 303 304 305 i
10.22.135.8/29 10.22.132.9 303 304 305 i
Total RIB entries = 30
```

# show ip cache

Use the **show ip cache** command to display the fast cache table.

## Syntax Description

No subcommands.

## Default Values

No default necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example shows sample output from the **show ip cache** command:

```
>enable
#show ip cache
DESTINATION     INTERFACE   NEXT HOP        USE COUNT   MAC ADDRESS
10.17.6.52      Loopback    127.0.0.1       231
172.22.77.80    eth 0/1     10.17.254.254   0           00:A0:C8:11:BA:32
10.17.255.255   Loopback    127.0.0.1       16
```

# show ip community-list

Use the **show ip community-list** command to display any or all defined community lists in the router configuration. Variations of this command include:

**show ip community-list**
**show ip community-list** <*name*>

## Syntax Description

| | |
|---|---|
| <*name*> | Optional. Specifies the name of the community list you wish to display. If this parameter is omitted, all defined community lists will be displayed. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example shows two community lists, one of which permits all routes containing community number 10:67, and another which permits routes containing community number 10:68 and the internet community number, but denies routes containing community number 10:45.

```
>enable
#show ip community-list
ip community-list CommList1:
 permit 10:67
ip community-list CommList2:
 permit 10:68 internet
 deny 10:45
```

# show ip dhcp-client lease

Use the **show ip dhcp-client lease** command to display all Dynamic Host Client Protocol (DHCP) lease information for interfaces that have dynamically assigned IP addresses. Variations of this command include:

**show ip dhcp-client lease**
**show ip dhcp-client lease** <*interface*>

## Syntax Description

| | |
|---|---|
| <*interface*> | Optional. Displays the information for the specified interface type. Specify an interface in the format <*interface type [slot/port* \| *slot/port.sub-interface id* \| *interface id* \| *interface id.sub-interface id]*>. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **show ip dhcp-client lease ?** for a complete list of applicable interfaces. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |

## Usage Examples

The following is a sample output from the **show dhcp-client lease** command:

>**enable**
**#show dhcp-client lease**
Interface: ethernet 0/1
Temp IP address: 10.100.23.64 Mask: 0.0.0.0
  DHCP Lease server: 10.100.23.207 State: Bound (3)
  Lease: 120 seconds
Temp default gateway address: 0.0.0.0
  Client-ID: N/A

# show ip dhcp-server binding

Use the **show ip dhcp-server binding** command to display the Dynamic Host Client Protocol (DHCP) server client table with associated information. Variations of this command include:

**show ip dhcp-server binding**
**show ip dhcp-server binding** <*ip address*>

## Syntax Description

| | |
|---|---|
| <*ip address*> | Optional. Specifies the IP address of the specified client. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |

## Usage Examples

The following is a sample output from the **show ip dhcp-server binding** command:

```
>enable
#show ip dhcp-server binding
IP Address      Client Id              Lease Expiration          Client Name
10.100.23.64    01:00:a0:c8:00:8f:b3   Aug 15 2002 11:02 AM      Router
```

# show ip ffe

Use the **show ip ffe** command to display all the FastFlow Engine (FFE) entries that match the specified parameters. Variations of this command include:

**show ip ffe destination** *<ip address>*
**show ip ffe destination-port** *<port>*
**show ip ffe details**
**show ip ffe egress** *<interface>*
**show ip ffe icmp-type** *<type>*
**show ip ffe ingress** *<interface>*
**show ip ffe protocol** *<protocol>*
**show ip ffe source** *<ip address>*
**show ip ffe source-port** *<port>*
**show ip ffe summary**
**show ip ffe type** *<type>*

The displayed output of these variations may be further defined through the use of additional subcommands. To view the additional subcommands available, type the variation of show command followed by **?**.

## Syntax Description

| | |
|---|---|
| **destination** | Displays FFE entries filtered on destination IP. |
| **destination-port** | Displays FFE entries filtered on destination port (TCP or UDP only). |
| **details** | Displays detailed information. |
| **egress** | Displays FFE entries filtered on egress interface. |
| **icmp-type** | Displays FFE entries filtered on ICMP type (ICMP only). |
| **ingress** | Displays FFE entries filtered on ingress interface. |
| **protocol** | Displays FFE entries filtered on protocol. |
| **source** | Displays FFE entries filtered on source IP. |
| **source-port** | Displays FFE entries filtered on source port (TCP or UDP only). |
| **summary** | Displays summary of FFE entries. |
| **type** | Displays FFE entries filtered on type. |
| *<ip address>* | Specifies a valid IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| *<interface>* | Specifies an interface in the format *<interface type [slot/port \| slot/port.sub-interface id \| interface id \| interface id.sub-interface id]>*. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **show ip ffe [egress \| ingress] ?** for a complete list of valid interfaces. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following is a sample output from the **show ip ffe summary** command:

```
>enable
#show ip ffe summary
Ingress         MaxEntries Entries     Hits    Misses     Drops
eth 0/1           4096         1        0        56         0
global           16384         1        0        56         0
```

# show ip igmp groups

Use the **show ip igmp groups** command to display the multicast groups that have been registered by directly connected receivers using Internet Group Management Protocol (IGMP). If no multicast group IP address is specified, all groups are shown with this command. Variations of this command include:

**show ip igmp groups**
**show ip igmp groups** *<multicast address>*

## Syntax Description

| | |
|---|---|
| *<multicast address>* | Optional. Displays the IP address of a multicast group. The multicast group IP address range is 244.0.0.0 to 239.255.255.255 or 224.0.0.0 /4. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |

## Usage Examples

The following is sample output from this command:

>**enable**
**#show ip igmp groups**
IGMP Connected Group Membership

| Group Address | Interface | Uptime | Expires | Last Reporter |
|---|---|---|---|---|
| 172.0.1.50 | Loopback100 | 00:42:57 | 00:02:50 | 172.23.23.1 |
| 172.1.1.1 | Ethernet0/1 | 00:05:26 | 00:02:51 | 1.1.1.2 |
| 172.1.1.1 | Loopback100 | 00:42:57 | 00:02:51 | 172.23.23.1 |

# show ip igmp interface

Use the **show ip igmp interface** command to display multicast-related information per-interface. If no interface is specified, this command shows information for all interfaces. Variations of this command include:

**show ip igmp interface**
**show ip igmp interface** *<interface>*

## Syntax Description

| | |
|---|---|
| *<interface>* | Displays information for a specific interface type. Specify an interface in the format *<interface type [slot/port \| slot/port.sub-interface id \| interface id \| interface id.sub-interface id]>*. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Enter the **show ip igmp interface ?** command for a complete list of interfaces. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |
| Release 9.1 | Command was expanded to include HDLC and tunnel interfaces. |

## Usage Examples

The following example is sample output from the **show ip igmp interface** command:

```
>enable
#show ip igmp interface
eth 0/1 is UP
 Ip Address is 10.22.120.47, netmask is 255.255.255.0
 IGMP is enabled on interface
 Current IGMP version is 2
 IGMP query interval is 60 seconds
 IGMP querier timeout is 120 seconds
 IGMP max query response time is 10 seconds
 Last member query count is 2
 Last member query response interval is 1000 ms
 IGMP activity: 548 joins, 0 leaves
 IGMP querying router is 0.0.0.0
 IGMP helper address is disabled
```

# show ip igmp snooping

Use the **show ip igmp snooping** command to display Internet Group Management Protocol (IGMP) snooping information. Variations of this command include:

**show ip igmp snooping**
**show ip igmp snooping mrouter**
**show ip igmp snooping mrouter vlan** <*vlan id*>
**show ip igmp snooping vlan**
**show ip igmp snooping vlan** <*vlan id*>

## Syntax Description

| | |
|---|---|
| **mrouter** | Optional. Displays the ports associated with multicast routers. |
| **vlan** <*vlan id*> | Optional. Displays whether IGMP snooping is enabled or disabled for a particular VLAN. (If global snooping is disabled, IGMP snooping cannot be enabled per VLAN. If global snooping is enabled, IGMP snooping can be enabled or disabled on a per VLAN basis.) |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following is sample output from the **ip igmp snooping vlan** command:

>**enable**
**#show ip igmp snooping vlan 1**
Vlan 1: IGMP snooping is enabled on this VLAN

The following is sample output from the **ip igmp snooping mrouter vlan** command:

>**enable**
**#show ip igmp snooping mrouter vlan 200**

| VLAN | Ports |
|---|---|
| 200 | Gi0/2(static) |

# show ip interfaces

Use the **show ip interfaces** command to display the status information for all IP interfaces (or a specific interface). Variations of this command include:

**show ip interfaces**
**show ip interfaces** <*interface*>
**show ip interfaces** <*interface*> **brief**

*To view secondary IP addresses, use the **show running-config** command.*

## Syntax Description

| | |
|---|---|
| <*interface*> | Optional. Displays status information for a specific interface. Specify an interface in the format <*interface type [slot/port | slot/port.sub-interface id | interface id | interface id.sub-interface id]*>. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **show ip interfaces ?** for a complete list of applicable interfaces. If no interface is specified, status information for all interfaces is displayed. |
| **brief** | Displays an abbreviated version of interface statistics for all IP interfaces. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 9.1 | Command was expanded to include HDLC and tunnel interfaces. |
| Release 11.1 | Demand interface was added. |

## Usage Examples

The following is a sample output of the **show ip interfaces** command:

```
>enable
#show ip interfaces
eth 0/1 is UP, line protocol is UP
  Ip address is 10.10.10.1
  Netmask is 255.255.255.0
  MTU is 1500
  Fastcaching is Enabled
  RIP Authentication is Disabled
 RIP Tx uses global version value
```

# show ip local policy

Use the **show ip local policy** command to display information about the route-map used for local policy-based routing.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following is sample output from this command:

```
>enable
#show ip local policy
Local policy routing is enabled, using route-map equal
route-map equal, permit, sequence 10
 Match clauses:
   length 150 200
 Set clauses:
   ip next-hop 10.10.11.254
 Policy routing matches: 0 packets, 0 bytes
route-map equal, permit, sequence 20
 Match clauses:
   ip address (access-lists): 101
 Set clauses:
   ip next-hop 10.10.11.14
 Policy routing matches: 2 packets, 172 bytes
```

# show ip mroute

Use the **show ip mroute** command to display IP multicasting routing table information. Variations of this command include:

**show ip mroute**
**show ip mroute all**
**show ip mroute** <*ip address*>
**show ip mroute** <*interface*>
**show ip mroute summary**

## Syntax Description

| | |
|---|---|
| **all** | Optional: Displays all multicast routes, including those not used to forward multicast traffic. |
| <*ip address*> | Optional. Displays IP address of a multicast group. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| <*interface*> | Optional. Specifies an interface in the format <*interface type [slot/port | slot/port.sub-interface id | interface id | interface id.sub-interface id]*>. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. |
| **summary** | Optional. Displays a single-line summary for each entry in the IP multicast routing table. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |
| Release 9.1 | Command was expanded to include HDLC and tunnel interfaces. |
| Release 11.1 | The All option was added. |

## Usage Examples

The following is sample output from the **show ip mroute all** command:

```
>enable
#show ip mroute all
IP Multicast Routing Table
Flags: S - Sparse, C - Connected, P - Pruned, J - Join SPT, T - SPT-bit Set,
F - Register, R - RP-bit Set
Timers: Uptime/Expires
(*, 225.1.0.1), 01:17:34/00:03:25, RP 192.168.0.254, Flags: SC
 Forwarding Entry: Yes
 Incoming interface: tunnel 2, RPF nbr 172.16.2.10
 Outgoing interface list:
   eth 0/1, Forward, 01:17:34/00:03:25
```

# show ip ospf

Use the **show ip ospf** command to display general information regarding open shortest path first (OSPF) processes.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Usage Examples

The following is a sample output from the **show ip ospf** command:

```
>enable
#show ip ospf
Summary of OSPF Process with ID: 192.2.72.101
 Supports only single Type Of Service routes (TOS 0)
 SPF delay timer: 5 seconds, Hold time between SPFs: 10 seconds
 LSA interval: 240 seconds
 Number of external LSAs: 0, Checksum Sum: 0x0
 Number of areas: 0, normal: 0, stub: 0, NSSA: 0
```

## show ip ospf database

Use the **show ip ospf database** command to display information from the open shortest path first (OSPF) database regarding a specific router. There are several variations of this command which you can use to obtain information about different OSPF link state advertisements. The variations are shown below:

**show ip ospf** <*area id*> **database**
**show ip ospf** <*area id*> **database adv-router** <*ip address*>
**show ip ospf** <*area id*> **database database-summary**
**show ip ospf** <*area id*> **database network**
**show ip ospf** <*area id*> **database network** <*link-state id*>
**show ip ospf** <*area id*> **database network** <*link-state id*> **adv-router** <*ip address*>
**show ip ospf** <*area id*> **database network adv-router** <*ip address*>
**show ip ospf** <*area id*> **database router**
**show ip ospf** <*area id*> **database router** <*link-state id*>
**show ip ospf** <*area id*> **database router** <*link-state id*> **adv-router** <*ip address*>
**show ip ospf** <*area id*> **database router adv-router** <*ip address*>
**show ip ospf** <*area id*> **database summary**
**show ip ospf** <*area id*> **database summary** <*link-state id*>
**show ip ospf** <*area id*> **database summary** <*link-state id*> **adv-router** <*ip address*>
**show ip ospf** <*area id*> **database summary adv-router** <*ip address*>
**show ip ospf database**
**show ip ospf database adv-router** <*ip address*>
**show ip ospf database database-summary**
**show ip ospf database external**
**show ip ospf database external** <*link-state id*>
**show ip ospf database external** <*link-state id*> **adv-router** <*ip address*>
**show ip ospf database external adv-router** <*ip address*>
**show ip ospf database network**
**show ip ospf database network** <*link-state id*>
**show ip ospf database network** <*link-state id*> **adv-router** <*ip address*>
**show ip ospf database network adv-router** <*ip address*>
**show ip ospf database router**
**show ip ospf database router** <*link-state id*>
**show ip ospf database router** <*link-state id*> **adv-router** <*ip address*>
**show ip ospf database router adv-router** <*ip address*>
**show ip ospf database summary**
**show ip ospf database summary** <*link-state id*>
**show ip ospf database summary** <*link-state id*> **adv-router** <*ip address*>
**show ip ospf database summary adv-router** <*ip address*>

## Syntax Description

| | |
|---|---|
| **adv-router** <*ip address*> | Optional. Displays information about link-state advertisements from the specified advertising router IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| <*area id*> | Optional. Specifies an OSPF area ID. Refer to *network* <*ip address*> <*wildcard mask*> *area* <*area id*> on page 1711 for more information. |
| **database** | Displays a detailed list of link-state advertisements for the specified area. |
| **database-summary** | Displays a simplified list of link-state advertisements for the specified area. |
| **external** | Displays information about external link-state advertisements from the specified link-state ID. |
| <*link-state id*> | Optional. Displays information from a specific link-state ID. The value defined in this field is tied to the advertisement's LS type. |
| **network** | Displays information about network link-state advertisements for the specified area or from the specified link-state ID. |
| **router** | Displays information about router link-state advertisements for the specified area or from the specified link-state ID. |
| **summary** | Displays information about summary link-state advertisements for the specified area or from the specified link-state ID. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Functional Notes

The link state ID differs depending on whether the link state advertisement in question describes a network or a router.

If describing a network, this ID is one of the following:

- The network's IP address. This is true for type 3 summary link advertisements and in autonomous system external link advertisements.
- An address obtained from the link state ID. If the network link advertisement's link state ID is masked with the network's subnet mask, this will yield the network's IP address.

If describing a router, this ID is always the router's OSPF router ID.

## Usage Examples

The following example shows the database link state summary for all areas:

```
>enable
#show ip ospf database
```

# show ip ospf interface

Use the **show ip ospf interface** command to display open shortest path first (OSPF) information for a specific interface. Variations of this command include:

**show ip ospf interface**
**show ip ospf interface** <*interface*>

## Syntax Description

| | |
|---|---|
| <*interface*> | Optional. Specifies an interface in the format <*interface type [slot/port | slot/port.sub-interface id | interface id | interface id.sub-interface id]*>. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**.Type **show ip ospf interface ?** for a complete list of applicable interfaces. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |
| Release 9.1 | Command was expanded to include HDLC and tunnel interfaces. |

## Usage Examples

The following example shows OSPF information for the PPP 1 interface.

>**enable**
**#show ip ospf interface ppp 1**

# show ip ospf neighbor

Use the **show ip ospf neighbor** command to display open shortest path first (OSPF) neighbor information for a specific interface. Variations of this command include:

**show ip ospf neighbor**
**show ip ospf neighbor detail**
**show ip ospf neighbor** *<interface>*
**show ip ospf neighbor** *<interface>* **detail**
**show ip ospf neighbor** *<interface>* *<neighbor id>*
**show ip ospf neighbor** *<interface>* *<neighbor id>* **detail**
**show ip ospf neighbor** *<neighbor id>*
**show ip ospf neighbor** *<neighbor id>* **detail**

## Syntax Description

| | |
|---|---|
| **detail** | Optional. Displays detailed information on neighbors. |
| *<interface>* | Optional. Specifies an interface in the format *<interface type [slot/port \| slot/port.sub-interface id \| interface id \| interface id.sub-interface id]>*. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **show ip ospf neighbor ?** for a complete list of applicable interfaces. |
| *<neighbor id>* | Optional. Specifies a specific neighbor's router ID. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |
| Release 9.1 | Command was expanded to include HDLC and tunnel interfaces. |

## Usage Examples

The following example shows detailed information on the OSPF neighbors:

```
>enable
#show ip ospf neighbor
```

# show ip ospf summary-address

Use the **show ip ospf summary-address** command to display a list of all summary address redistribution information for the system.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Usage Examples

The following example displays all summary address redistribution information for the system:

```
>enable
#show ip ospf summary-address
```

# show ip pim-sparse

Use the **show ip pim-sparse** command to display Protocol Independent Multicast (PIM) configuration information. Sparse mode or PIM-SM is a routing protocol used to establish and maintain the multicast distribution tree. Routers can participate in the shared tree (RPT) rooted at the rendezvous point (RP) router or the shortest-path tree (SPT) rooted at a multicast source. PIM-SM also establishes both shared trees and shortest-path trees. Variations of this command include:

**show ip pim-sparse**
**show ip pim-sparse interfaces** <*interface*>
**show ip pim-sparse neighbor**
**show ip pim-sparse rp-map**
**show ip pim-sparse rp-set**
**show ip pim-sparse state**
**show ip pim-sparse traffic**

## Syntax Description

| | |
|---|---|
| **interfaces** <*interface*> | Displays PIM-SM configuration and status information for a specific interface. Specify an interface in the format <*interface type [slot/port | slot/port.sub-interface id | interface id | interface id.sub-interface id]*>. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **show ip pim-sparse interface ?** to display a list of applicable interfaces. |
| **neighbor** | Displays neighbor adjacency information. |
| **rp-map** | Displays active group-to-RP mappings. |
| **rp-set** | Displays a list of statically configured RP candidates. The multicast group IP address is 224.0.0.0 /4 when no access group was applied to the **rp-address** command (refer to *rp-address <ip address>* on page 1720). Otherwise it is the name of the access group. |
| **state** | Displays multicast route PIM state information. |
| **traffic** | Displays active PIM-SM control traffic statistics. |

## Default Values

No default necessary for this command.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example shows sample output from the **show ip pim-sparse** command:

```
>enable
#show ip pim-sparse
Global PIM Sparse Mode Settings
 Join/Prune interval: 60, SPT threshold: 1
```

The following example shows sample output from the **show ip pim-sparse interface** command:

```
>enable
#show ip pim-sparse interface
eth 0/1 is UP
  PIM Sparse
  DR: itself
  Local Address: 192.168.1.254
  Hello interval (sec): 30, Neighbor timeout (sec): 105
  Propagation delay (ms): 500, Override interval (ms): 2500

tunnel 1 is UP
  PIM Sparse
  DR: 172.16.1.10
  Local Address: 172.16.1.9
  Hello interval (sec): 30, Neighbor timeout (sec): 105
  Propagation delay (ms): 500, Override interval (ms): 2500

tunnel 2 is UP
  PIM Sparse
  DR: 172.16.2.10
  Local Address: 172.16.2.9
  Hello interval (sec): 30, Neighbor timeout (sec): 105
  Propagation delay (ms): 500, Override interval (ms): 2500
```

The following example shows sample output from the **show ip pim-sparse neighbor** command:

```
>enable
#show ip pim-sparse neighbor
Port          Neighbor       Holdtime(sec)  Age(sec)      Uptime(sec)
-------------------------------------------------------------------------------------------
tunnel 1      172.16.1.10    105            19            241908
tunnel 2      172.16.2.10    105            23            241913
```

The following example shows sample output from the **show ip pim-sparse rp-map** command:

```
>enable
#show ip pim-sparse rp-map
Number of group-to-RP mappings: 5
Group address           RP address
-------------------------------------------------------
225.1.0.1               192.168.0.254
225.1.0.2               192.168.0.254
225.1.0.3               192.168.0.254
```

The following example shows sample output from the **show ip pim-sparse rp-map set** command:

```
>enable
#show ip pim rp-map set
Group address           Static-RP-address
-------------------------------------------------------------
224.0.0.0/4             192.168.0.254
MCAST_ACL_1             192.168.1.254
MCAST_ACL_2             192.168.2.254
MCAST_ACL_3             192.168.3.254
```

The following example shows sample output from the **show ip pim-sparse state** command:

```
>enable
#show ip pim-sparse state
PIM-SM State Table
Flags: S - Sparse, C - Connected, P - Pruned, J - Join SPT, T - SPT-bit Set,
F - Register, R - RP-bit Set
Timers: Uptime/Expires

(*, 225.1.0.1), 02:42:03/00:03:04, RP 192.168.0.254, Flags: SC
 Forwarding Entry: Yes
 Incoming interface: tunnel 2, RPF nbr 172.16.2.10
 Upstream Join/Prune State: Joined
 Register State: No Info
 RegStop Timer (sec): stopped
 Join/Prune Timer (sec): 57
 Override Timer (sec): stopped
 Multicast Border Router: 0.0.0.0
 Packets Forwarded: 2
 Outgoing interface list:
   eth 0/1, Forward, 02:42:03/00:03:03
     Downstream Join/Prune State: Join
     Assert Winner State: No Info
     Assert Timer (sec): stopped
     Assert Winner: 0.0.0.0
```

```
    Assert Winner Metric: infinity
    Local Membership: Yes
    Forwarding State: Forwarding
Inherited output list:
  eth 0/1
```

The following example shows sample output from the **show ip pim-sparse traffic** command:

```
>enable
#show ip pim-sparse traffic
                Rx      Tx      Rx              Tx
Port: eth 0/1
 Hello:         7       8334    J/P:    0       0
 Register:      0       0       RegStop:0       0
 Assert:        0       0
Port: tunnel 1
 Hello:         8327    8333    J/P:    0       57
 Register:      0       0       RegStop:0       0
 Assert:        0       0
Port: tunnel 2
 Hello:         8323    8334    J/P:    0       11949
 Register:      0       0       RegStop:0 0
 Assert:        0       0
Total
 Hello:         16657   25001   J/P:    0       12006
 Register:      0       0       RegStop:0       0
 Assert:        0       0
```

# show ip policy

Use the **show ip policy** command to display the interfaces which have route maps configured. This command is used for troubleshooting policy-based routing.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following is sample output from this command:

```
>enable
#show ip policy

Interface       Route-map
eth 0/1         ISP_A
eth 0/2         ISP_B
```

# show ip policy-class

Use the **show ip policy-class** command to display the configured session limit and specific host IP addresses of all current sessions. Refer to *ip policy-class <name>* on page 549 for information on configuring access policies.Variations of this command include:

**show ip policy-class**
**show ip policy-class** *<name>*
**show ip policy-class host-sessions**
**show ip policy-class** *<name>* **host-sessions**

## Syntax Description

| | |
|---|---|
| **host-sessions** | Optional. Displays specific host IP addresses of all current sessions |
| *<name>* | Optional. Displays policy class information for a specific policy class. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |
| Release 12.1 | Command expanded to include **host-sessions**. |

## Usage Examples

The following is a sample output from the **show ip policy-class** command:

```
>enable
#show ip policy-class

Maximum policy-sessions: 17400

Policy-class “Private”:
 136 current sessions (5800 max)
 Entry 1 - allow list self self
 Entry 2 - nat source list wizard-ics interface ppp 1 overload

Policy-class “Public”:
 0 current sessions (5800 max)
```

The following is a sample output from the **show ip policy-class host-sessions** command:

**>enable**
**#show ip policy-class host-sessions**

```
Policy-class “Private”:
 100 policy-sessions allowed per source address.

 Src IP Address   Sessions
 ---------------- --------
 192.168.1.100    1
 192.168.1.101    35
 192.168.1.121    100   (maximum allowed)

Policy-class “Public”:
 No limit for policy-sessions allowed per host.
```

The following is a sample output from the **show ip policy-class** <*policyname*> **host-sessions** command for the policy class named **Private**:

>**enable**
**#show ip policy-class Private host-sessions**

```
Policy-class “Private”:
 100 policy-sessions allowed per source address.

 Src IP Address   Sessions
 ---------------- --------
 192.168.1.100    1
 192.168.1.101    35
 192.168.1.121    100   (maximum allowed)
```

# show ip policy-sessions

Use the **show ip policy-sessions** command to display a list of current policy class associations. Refer to *ip policy-class <name>* on page 549 for information on configuring access policies. Variations of this command include:

**show ip policy-sessions**
**show ip policy-sessions all**
**show ip policy-sessions** *<name>*
**show ip policy-sessions** *<name>* **all**

## Syntax Description

| | |
|---|---|
| *<name>* | Optional. Displays policy class associations for the specified policy class. |
| **all** | Optional. Displays all policy sessions, including active associations (through which the firewall is allowed to pass traffic) and associations flagged for deletion (through which the firewall is forbidden to pass traffic). Associations flagged for deletion will usually be freed within a few seconds of timeout or deletion, depending on packet congestion; servicing of packets is given priority. New traffic matching an association will create a new active association, provided the traffic still matches a policy-class **allow** or **NAT** entry. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |
| Release 11.1 | Command expanded to include the **all** option. |

## Usage Examples

The following is sample output from the **show ip policy-sessions** command:

```
>enable
#show ip policy-sessions
Protocol (TTL) [in crypto map] -> [out crypto map] Destination policy-class
Src IP Address      Src Port   Dest IP Address     Dst Port   NAT IP Address   NAT Port
-------------------------------------------------------------------------------------------------------------------------------
Policy class “Public”:
tcp (13)
 192.168.1.142      2621       192.168.19.2        1          10.10.10.1       3000
tcp (13)
 192.168.1.142      2622       192.168.19.2        2          10.10.10.1       3001
tcp (13)
 192.168.1.142      2623       192.168.19.2        3          10.10.10.1       3002
```

The following is sample output from the **show ip policy-sessions all** command:

>**enable**
**#show ip policy-sessions all**

Protocol (TTL) [in crypto map] -> [out crypto map] Destination policy-class

| Src IP Address | Src Port | Dest IP Address | Dst Port | NAT IP Address | NAT Port |
|---|---|---|---|---|---|
| Policy class “Public”: | | | | | |
| tcp (0) - inactive | | | | | |
| 192.168.1.142 | 1025 | 192.168.19.2 | 3135 | 10.10.10.1 | 3605 |
| tcp (0) - inactive | | | | | |
| 192.168.1.142 | 1028 | 192.168.19.2 | 3138 | 10.10.10.1 | 3606 |
| tcp (0) - inactive | | | | | |
| 192.168.1.142 | 1029 | 192.168.19.2 | 3139 | 10.10.10.1 | 3607 |
| tcp (0) - inactive | | | | | |
| 192.168.1.142 | 1036 | 192.168.19.2 | 3146 | 10.10.10.1 | 3608 |

# show ip policy-stats

Use the **show ip policy-stats** command to display a list of current policy class statistics. Refer to *ip policy-class <name>* on page 549 for information on configuring access policies. Variations of this command include:

**show ip policy-stats**
**show ip policy-stats** *<name>*

## Syntax Description

| | |
|---|---|
| *<name>* | Optional. Displays policy class statistics for a specific policy class. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Usage Examples

The following example displays a list of current policy class statistics:

```
>enable
#show ip policy-stats
```

# show ip prefix-list

Use the **show ip prefix-list** command to display BGP prefix list information. Variations of this command include:

**show ip prefix-list** <*name*>
**show ip prefix-list detail** <*name*>
**show ip prefix-list summary** <*name*>

## Syntax Description

| | |
|---|---|
| <*name*> | Shows information for a specific prefix list. |
| **detail** | Optional. Shows a listing of the prefix list rules and their hit counts. |
| **summary** | Optional. Shows information about the entire prefix list. |

## Default Values

No default values are necessary for this command.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Functional Notes

If the **show ip prefix-list** command is issued with no arguments, a listing of the prefix-list rules but no hit count statistics is displayed.

## Usage Examples

The following example displays information about the prefix list **test**.

```
>enable
#show ip prefix-list test
ip prefix-list test: 4 entries
  seq 5 permit 0.0.0.0/0 ge 8 le 8
  seq 10 deny 0.0.0.0/0 ge 9 le 9
  seq 15 permit 0.0.0.0/0 ge 10 le 10
  seq 20 deny 0.0.0.0/0 ge 11
```

# show ip protocols

Use the **show ip protocols** command to display IP routing protocol parameters and statistics.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Usage Examples

The following is a sample output from the **show ip protocols** command:

```
>enable
#show ip protocols
Sending updates every 30 seconds, next due in 8 seconds
 Invalid after 180 seconds, hold down time is 120 seconds
 Redistributing: rip
 Default version control: send version 2, receive version 2
   Interface        Send Ver.        Rec Ver.
    eth 0/1             2                2
     ppp 1              2                2
 Routing for networks:
   1.1.1.0/24
```

# show ip route

Use the **show ip route** command to display the contents of the IP route table. Variations of this command include:

**show ip route**
**show ip route** *<ip address>*
**show ip route** *<ip address> <subnet mask>*
**show ip route bgp**
**show ip route connected**
**show ip route ospf**
**show ip route rip**
**show ip route static**
**show ip route summary**
**show ip route summary realtime**
**show ip route table**

*Using the **realtime** argument for this command can adversely affect the performance or your unit.*

## Syntax Description

| | |
|---|---|
| *<ip address>* | Optional. Specifies a valid IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| *<subnet mask>* | Optional. Specifies the subnet mask that corresponds to a range of IP addresses (network) or a specific host. Subnet masks can be expressed in dotted decimal notation (for example, **255.255.255.0**) or as a prefix length (for example, **/24**). |
| **bgp** | Displays only the IP routes associated with BGP. |
| **connected** | Optional. Displays only the IP routes for directly connected networks. |
| **ospf** | Optional. Displays only the IP routes associated with OSPF. |
| **rip** | Optional. Displays only the IP routes that were dynamically learned through RIP. |
| **static** | Optional. Displays only the IP routes that were statically entered. |
| **summary** | Optional. Displays a summary of all IP route information. |
| **summary realtime** | Optional. Displays full-screen output in real time. Refer to the *Functional Notes* below for more information. |
| **table** | Optional. Displays a condensed version of the IP route table. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 10.1 | The real time display option was introduced. |

## Functional Notes

Use the **realtime** argument for this command to display full-screen output in real time. Information is continuously updated on the console until you either freeze the data (by pressing the **F** key) or exit **realtime** mode (by pressing **Ctrl-C**). If there is not enough room on the screen for all available data, the information will truncate at the bottom of the screen. In order to maximize the amount of data displayed, increase the terminal length (using the **terminal length** command; refer to *terminal length <number>* on page 410).

## Usage Examples

The following is a sample output from the **show ip route** command:

```
>enable
#show ip route rip
Codes: C - connected S - static R - RIP O - OSPF IA - OSPF inter area
       N1 - OSPF NSSA external type 1 N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1 E2 - OSPF external type 2
Gateway of last resort is 10.200.254.254 to network 0.0.0.0
```

The following example shows how to display IP routes learned via BGP. The values in brackets after a BGP route entry represent the entry's administrative distance and metric:

```
>enable
#show ip route bgp
Codes: C - connected, S - static, R - RIP, O - OSPF, B - BGP
       IA - OSPF inter area, N1 - OSPF NSSA external type 1
       N2 - OSPF NSSA external type 2, E1 - OSPF external type 1
       E2 - OSPF external type 2
Gateway of last resort is 10.15.43.17 to network 0.0.0.0
B    1.0.0.0/8 [30/0] via 10.15.43.17, fr 1.17
B    2.0.0.0/9 [30/0] via 10.15.43.17, fr 1.17
B    2.128.0.0/10 [30/0] via 10.15.43.17, fr 1.17
B    2.192.0.0/11 [30/0] via 10.15.43.17, fr 1.17
B    2.224.0.0/12 [30/0] via 10.15.43.17, fr 1.17
B    2.240.0.0/13 [30/0] via 10.15.43.17, fr 1.17
B    2.248.0.0/14 [30/0] via 10.15.43.17, fr 1.17
```

# show ip traffic

Use the **show ip traffic** command to display all IP traffic statistics. Variations of this command include:

**show ip traffic**
**show ip traffic realtime**


*Using the **realtime** argument for this command can adversely affect the performance or your unit.*

## Syntax Description

| | |
|---|---|
| **realtime** | Optional. Displays full-screen output in real time. Refer to the *Functional Notes* below for more information. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |
| Release 10.1 | The real time display option was introduced. |

## Functional Notes

Use the **realtime** argument for this command to display full-screen output in real time. Information is continuously updated on the console until you either freeze the data (by pressing the **F** key) or exit **realtime** mode (by pressing **Ctrl-C**). If there is not enough room on the screen for all available data, the information will truncate at the bottom of the screen. In order to maximize the amount of data displayed, increase the terminal length (using the **terminal length** command; refer to *terminal length <number>* on page 410).

## Usage Examples

The following example displays all IP traffic statistics:

>**enable**
**#show ip traffic**

IP statistics:
Routing discards: 0
Rcvd: 15873 total, 7617 delivered
0 header errors, 0 address errors
0 unknown protocol, 0 discards
0 checksum errors, 0 bad hop counts
Sent: 8281 generated, 4459 forwarded
0 no routes, 0 discards
Frags: 0 reassemble required, 0 reassembled, 0 couldn't reassemble
0 created, 0 fragmented, 0 couldn't fragment

UDP statistics:
Rcvd: 3822 total, 0 checksum errors, 0 no port
Sent: 3822 total

TCP statistics:
Retrans Timeout Algorithm: 0
Min retrans timeout (ms): 0
Max retrans timeout (ms): 0
Max TCP Connections: 0
0 active opens, 64 passive opens, 0 failed attempts
5 establish resets, 1 establish current
3795 segments received, 4459 segments sent, 26 segments retransmitted

# show ip urlfilter

Use the **show ip urlfilter** command to display configured URL filter and server information.

## Syntax Description

No subcommands.

## Default Values

No default necessary for this command.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example shows sample output from the **show ip urlfilter** command:

```
>enable
#show ip urlfilter

Configured for Websense URL filtering.
Filters
-------
Name: “filter1”
Ports: HTTP(80)
Interfaces that filter is applied to:
eth 0/2 inbound
Servers
-------
IP address: 10.100.23.116
Port: 15868
Timeout: 5
Excluded domains
----------------
Permit www.adtran.com
Other Settings
--------------
Allow mode: Off
Maximum outstanding requests: 500
Maximum number of response packets buffered: 100
```

# show ip urlfilter exclusive-domain

Use the **show ip urlfilter exclusive-domain** to display all configured domains excluded (either always allowed or always blocked) from URL filtering.

## Syntax Description

No subcommands.

## Default Values

No default necessary for this command.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example shows sample output from the **show ip urlfilter exclusive-domain** command:

```
>enable
#show ip urlfilter exclusive-domain

Excluded domains
----------
Permit www.adtran.com
```

# show ip urlfilter statistics

Use the **show ip urlfilter statistics** command to display statistics for URL filter requests and responses.

## Syntax Description

No subcommands.

## Default Values

No default necessary for this command.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example shows sample output from the **show ip urlfilter statistics** command:

```
>enable
#show ip urlfilter statistics

Current outstanding requests to filter server: 0
Current response packets buffered from web server: 2
Max outstanding requests to filter server: 3
Max response packets buffered from web server: 5
Total requests sent to filter server: 400
Total responses received from filter server: 400
Total requests allowed: 398Total requests blocked: 2
```

# show isdn-group <*number*>

Use the **show isdn group** command to display integrated services digital network (ISDN) group information.

## Syntax Description

| | |
|---|---|
| <*number*> | Displays information for a specific ISDN group. Valid range is 1 to 255. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example displays information for ISDN group **5**:

```
>enable
#show isdn-group 5
```

# show isdn-number-template <*value*>

Use the **show isdn-number-template** command to display integrated services digital network (ISDN) number templates. Variations of this command include:

**show isdn-number-template**
**show isdn-number-template** <*value*>

## Syntax Description

| | |
|---|---|
| <*value*> | Optional. Displays information about a specific number template. Valid range is 1 to 255. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example displays information for ISDN number template **0**:

```
>enable
#show isdn-number-template 0
Type          ID      Prefix    Pattern
Subscriber    0                 911
#
```

# show isdn resource

Use the **show isdn resource** command to display integrated services digital network (ISDN) resource information. Variations of this command include:

**show isdn resource**
**show isdn resource realtime**


*Using the **realtime** argument for this command can adversely affect the performance or your unit.*

## Syntax Description

| | |
|---|---|
| **realtime** | Optional. Displays full-screen output in real time. Refer to the *Functional Notes* below for more information. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Functional Notes

Use the **realtime** argument for this command to display full-screen output in real time. Information is continuously updated on the console until you either freeze the data (by pressing the **F** key) or exit **realtime** mode (by pressing **Ctrl-C**). If there is not enough room on the screen for all available data, the information will truncate at the bottom of the screen. In order to maximize the amount of data displayed, increase the terminal length (using the **terminal length** command; refer to *terminal length <number>* on page 410).

## Usage Examples

The following example displays ISDN resource information:

**>enable**
**#show isdn resource**

| Interface: ChannelId | Channel State:GID | Trunk: Appearance | Appearance State | Slot/Prt: B-Channel | Call State |
|---|---|---|---|---|---|
| pri 1:0 | Reserved:1 | T01:2 | TAS_Connecte | 1/1:21 | OutgoingConnect |
| pri 1:1 | Reserved:1 | T01:0 | TAS_Alerting | 1/1:23 | IncomingAlertingSent |
| pri 1:2 | Available | --- | --- | --- | --- |
| pri 1:3 | Available | --- | --- | --- | --- |
| pri 1:4 | Available | --- | --- | --- | --- |
| pri 1:5 | Available | --- | --- | --- | --- |
| pri 1:6 | Available | --- | --- | --- | --- |
| pri 1:7 | Available | --- | --- | --- | --- |
| pri 1:8 | Available | --- | --- | --- | --- |
| pri 1:9 | Available | --- | --- | --- | --- |
| pri 1:10 | Available | --- | --- | --- | --- |
| pri 1:11 | Available | --- | --- | --- | --- |
| pri 1:12 | Available | --- | --- | --- | --- |
| pri 1:13 | Available | --- | --- | --- | --- |
| pri 1:14 | Available | --- | --- | --- | --- |
| pri 1:15 | Available | --- | --- | --- | --- |
| pri 1:16 | Available | --- | --- | --- | --- |
| pri 1:17 | Available | --- | --- | --- | --- |
| pri 1:18 | Available | --- | --- | --- | --- |
| pri 1:19 | Available | --- | --- | --- | --- |
| pri 1:20 | Available | --- | --- | --- | --- |
| pri 1:21 | Available | --- | --- | --- | --- |
| pri 1:22 | Available | --- | --- | --- | --- |

# show lldp

Use the **show lldp** command to display local loop demarkation point (LLDP) timer configuration.

## Syntax Description

No subcommands.

## Default Values

No default values are necessary for this command.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Usage Examples

The following example shows a sample LLDP timer configuration:

```
>enable
#show lldp
Global LLDP information:
Sending LLDP packets every 30 seconds
Sending TTL of 120 seconds
```

# show lldp device *<name>*

Use the **show lldp device** command to display specific neighbor information about a given neighbor.

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies the system name of the neighbor to display. |

## Default Values

No default values are necessary for this command.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Functional Notes

If there is more than one neighbor with the same system name, all neighbors with that system name will be displayed.

## Usage Examples

The following example shows specific information about a neighbor for the system name **Router**:

>**enable**
**#show lldp device Router**
Chassis ID: 00:A0:C8:02:DD:2A (MAC Address)
System Name: Router
Device Port: eth 0/1 (Locally Assigned)
Holdtime: 30
Platform: NetVanta 3305
Software: Version: 08.00.22.sw1.D, Date: Mon Nov 01 10:28:55 2004
Capabilities: Bridge, Router
Enabled Capabilities: Router
Local Port: eth 0/3
Management Addresses:
Address Type: IP version 4, Address: 10.23.10.10
Interface Type: Interface Index, Interface Id: 2

# show lldp interface *<interface>*

Use the **show lldp interface** command to display local loop demarkation point (LLDP) configuration and statistics for interfaces on this device.

## Syntax Description

| | |
|---|---|
| *<interface>* | Optional. Displays the information for the specified interface. Specify an interface in the format *<interface type [slot/port \| slot/port.sub-interface id \| interface id \| interface id.sub-interface id]>*. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **show lldp interface ?** for a complete list of applicable interfaces. |

## Default Values

No default values are necessary for this command.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Usage Examples

The following example shows LLDP configuration and statistics for the Ethernet 0/1 interface:

```
>enable
#show lldp interface ethernet 0/1
eth 0/1 (TX/RX)
    0 packets input
    0 input errors
    0 TLV errors, 0 TLVs Discarded
    0 packets discarded
    8799 packets output
    0 neighbor ageouts
#
```

# show lldp neighbors

Use the **show lldp neighbors** command to display information about neighbors of this device learned about via local loop demarkation point (LLDP). Variations of this command include:

**show lldp neighbors**
**show lldp neighbors detail**
**show lldp neighbors interface** <*interface*>
**show lldp neighbors realtime**

*Using the **realtime** argument for this command can adversely affect the performance or your unit.*

## Syntax Description

| | |
|---|---|
| **detail** | Optional. Shows detailed neighbor information for the specified interface or interface type. |
| **interface** <*interface*> | Optional. Displays a summary of all neighbors learned about through the specified interface. Specify an interface in the format <*interface type [slot/port | slot/port.sub-interface id | interface id | interface id.sub-interface id]*>. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **show lldp neighbors interface ?** for a complete list of applicable interfaces. |
| **realtime** | Optional. Displays full-screen output in real time. Refer to the *Functional Notes* below for more information. |

## Default Values

No default values necessary for this command.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |
| Release 10.1 | The real time display option was introduced. |

## Functional Notes

Use the **realtime** argument for this command to display full-screen output in real time. Information is continuously updated on the console until you either freeze the data (by pressing the **F** key) or exit **realtime** mode (by pressing **Ctrl-C**). If there is not enough room on the screen for all available data, the information will truncate at the bottom of the screen. In order to maximize the amount of data displayed, increase the terminal length (using the **terminal length** command; refer to *terminal length <number>* on page 410).

## Usage Examples

The following example shows detailed information about a device's neighbors:

```
>enable
#show lldp neighbors interface eth 0/3 detail
Chassis ID: 00:A0:C8:02:DD:2A (MAC Address)
 System Name: Router
 Device Port: eth 0/1 (Locally Assigned)
 Holdtime: 38
 Platform: NetVanta 3305
  Software: Version: 08.00.22.sw1.D, Date: Mon Nov 01 10:28:55 2004
  Capabilities: Bridge, Router
  Enabled Capabilities: Router
 Local Port: eth 0/3
 Management Addresses:
   Address Type: IP version 4, Address: 10.23.10.10
Interface Type: Interface Index, Interface Id: 2
```

# show lldp neighbors statistics

Use the **show lldp neighbors statistics** command to display statistics about local loop demarkation point (LLDP) neighbor table actions.

## Syntax Description

No subcommands.

## Default Values

There are no default values necessary for this command.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Functional Notes

This command shows information about the changes in this device's neighbor table. The information displayed indicates the last time a neighbor was added to or removed from the table as well as the number of times neighbors were inserted into or deleted from the table.

| | |
|---|---|
| **System Last Change Time** | Shows the time at which the most recent change occurred in the neighbor table. |
| **Inserts** | Shows the number of times neighbors have been added to the table. |
| **Deletes** | Shows how many times neighbors have been deleted from the table because an interface was shut down. |
| **Drops** | Shows how many times the insertion of a new neighbor into the table failed because the table was full. |
| **Age Outs** | Shows how many times neighbors have been removed from the table because no new updates were received from that neighbor before its time-to-live timer expired. |

## Usage Examples

The following example shows sample output for this command:

```
>enable
#show lldp neighbors statistics
System Last Change Time   Inserts   Deletes   Drops   Age Outs
10-15-2004 14:24:56       55        3         1       1
```

# show mac address-table

Use the **show mac address-table** command to display all static and dynamic entries in the medium access control (MAC) address table for all virtual local area networks (VLANs) and physical interfaces.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following is sample output from the **show mac address-table** command:

```
>enable
#show mac address-table
Mac Address Table
------------------------------------------
Vlan    Mac Address         Type        Ports
1       aa:bb:ee:d1:c2:33   STATIC      eth 0/18
1       00:00:00:00:00:00   STATIC      CPU
2       00:90:2b:7d:30:00   DYNAMIC     eth 0/1
2       00:a0:c8:00:8e:a6   DYNAMIC     eth 0/1
2       00:a0:c8:00:8f:ba   DYNAMIC     eth 0/1
2       00:a0:c8:00:8f:73   DYNAMIC     eth 0/1
2       00:a0:c8:00:00:00   DYNAMIC     eth 0/1
2       00:a0:c8:01:ff:02   DYNAMIC     eth 0/1
2       00:a0:c8:01:09:d3   DYNAMIC     eth 0/1
2       00:a0:c8:01:13:34   DYNAMIC     eth 0/1
2       00:a0:c8:01:14:4a   DYNAMIC     eth 0/1
2       00:a0:c8:03:95:4b   DYNAMIC     eth 0/1
2       00:a0:c8:05:00:89   DYNAMIC     eth 0/1
2       00:a0:c8:05:00:ac   DYNAMIC     eth 0/1
2       00:a0:c8:05:00:ad   DYNAMIC     eth 0/1
2       00:a0:c8:05:00:c2   DYNAMIC     eth 0/1
```

# show mac address-table address

Use the **show mac address-table address** command to display all medium access control (MAC) addresses known by AOS. Variations of this command include the following:

**show mac address-table address** <*mac address*>
**show mac address-table address** <*mac address*> **interface** <*interface*>
**show mac address-table address** <*mac address*> **interface** <*interface*> **vlan** <*vlan id*>
**show mac address-table address** <*mac address*> **vlan** <*vlan id*>

## Syntax Description

| | |
|---|---|
| <*mac address*> | Specifies a valid 48-bit MAC address. MAC addresses should be expressed in following format xx:xx:xx:xx:xx:xx (for example, **00:A0:C8:00:00:01**). |
| **interface** <*interface*> | Shows information for a specific interface. Specify an interface in the format <*interface type [slot/port* \| *slot/port.sub-interface id* \| *interface id* \| *interface id.sub-interface id]*>. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **show mac address-table address interface ?** for a list of valid interfaces. |
| **vlan** <*vlan id*> | Specifies a valid VLAN interface ID. Range is 1 to 4094. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following sample from the **show mac address-table address** command displays information regarding a specific MAC address from the MAC address table:

```
>enable
#show mac address-table address 00:a0:c8:7d:30:00
Mac Address Table
------------------------------------------
Vlan    Mac Address         Type        Ports
--------  -------------------  ---------  -----------
2       00:a0:c8:7d:30:00   DYNAMIC     eth 0/1
```

The following sample from the **show mac address-table address** command displays information regarding a specific MAC address and interface from the MAC address table:

```
>enable
#show mac address-table address 00:a0:c8:7d:30:00 ethernet 0/1
Mac Address Table
-------------------------------------------
Vlan    Mac Address          Type         Ports
--------  -------------------  ---------  -----------
2       00:a0:c8:7d:30:00    DYNAMIC    eth 0/1
Total Mac Addresses for this criterion: 1
#
```

# show mac address-table aging-time

Use the **show mac address-table aging-time** command to display information regarding the amount of time dynamic entries remain in the medium access control (MAC) address table.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following is a sample from the **show mac address-table aging-time** command for a switch configured with an address-table aging-time:

```
>enable
#show mac address-table aging-time
Aging Time
----------
300 Seconds
```

# show mac address-table count

Use the **show mac address-table count** command to display information regarding the number of medium access control (MAC) addresses in use (both static and dynamic).

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following is a sample from the **show mac address-table count** command:

```
>enable
#show mac address-table count
Mac Table Entries:
--------------------------
Dynamic Address Count: 19
Static Address Count: 3
Total Mac Addresses: 23
Total Mac Address Space Available: 8169
```

# show mac address-table dynamic

Use the **show mac address-table dynamic** command to display all dynamic medium access control (MAC) addresses learned by AOS. Variations of this command include the following:

**show mac address-table dynamic**
**show mac address-table dynamic address** <*mac address*>
**show mac address-table dynamic address** <*mac address*> **interface** <*interface*>
**show mac address-table dynamic address** <*mac address*> **interface** <*interface*> **vlan** <*vlan id*>
**show mac address-table dynamic address** <*mac address*> **vlan** <*vlan id*>
**show mac address-table dynamic interface** <*interface*>
**show mac address-table dynamic interface** <*interface*> **vlan** <*vlan id*>
**show mac address-table dynamic vlan** <*vlan id*>

## Syntax Description

| | |
|---|---|
| **address** <*mac address*> | <*mac address*>Specifies a valid 48-bit MAC address. MAC addresses should be expressed in following format xx:xx:xx:xx:xx:xx (for example, **00:A0:C8:00:00:01**). |
| **interface** <*interface*> | Shows information for a specific interface. Specify an interface in the format <*interface type [slot/port* \| *slot/port.sub-interface id* \| *interface id* \| *interface id.sub-interface id]*>. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **show mac address-table dynamic interface ?** for a list of valid interfaces. |
| **vlan** <*vlan id*> | Specifies a valid VLAN interface ID. Range is 1 to 4094. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following is a sample from the **show mac address-table dynamic** command:

>**enable**
**#show mac address-table dynamic**
Mac Address Table

| Vlan | Mac Address | Type | Ports |
|---|---|---|---|
| 1 | 00:a0:c8:7d:30:00 | DYNAMIC | eth 0/1 |
| 1 | 00:a0:c8:05:89:09 | DYNAMIC | eth 0/2 |
| 1 | 00:a0:c8:07:d9:d2 | DYNAMIC | eth 0/5 |
| 1 | 00:a0:c8:07:d9:19 | DYNAMIC | eth 0/7 |
| 1 | 00:a0:c8:09:95:6b | DYNAMIC | eth 0/7 |
| 1 | 00:a0:c8:0a:2d:7c | DYNAMIC | eth 0/12 |
| 1 | 00:a0:c8:f6:e9:a6 | DYNAMIC | eth 0/24 |
| 1 | 00:a0:c8:01:0a:ef | DYNAMIC | eth 0/23 |
| 1 | 00:a0:c8:0c:74:80 | DYNAMIC | eth 0/20 |
| 1 | 00:a0:c8:15:5a:9f | DYNAMIC | eth 0/7 |
| 1 | 00:a0:c8:6c:71:49 | DYNAMIC | eth 0/2 |
| 1 | 00:a0:c8:77:78:c1 | DYNAMIC | eth 0/3 |
| 1 | 00:a0:c8:6b:53:7b | DYNAMIC | eth 0/4 |
| 1 | 00:a0:c8:72:e6:d6 | DYNAMIC | giga-eth 0/2 |
| 1 | 00:a0:c8:05:00:e6 | DYNAMIC | giga-eth 0/1 |

Total Mac Addresses for this criterion: 15

# show mac address-table interface

Use the **show mac address-table interface** command to display information regarding medium access control (MAC) address table entries specific to a certain interface. Variations of this command include:

**show mac address-table interface** <*interface*>
**show mac address-table interface** <*interface*> **vlan** <*vlan id*>

## Syntax Description

| | |
|---|---|
| <*interface*> | Shows information for a specific interface type. Specify an interface in the format <*interface type [slot/port | slot/port.sub-interface id | interface id | interface id.sub-interface id]*>. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **show mac address-table interface ?** for a list of valid interfaces. |
| **vlan** <*vlan id*> | Optional. Shows address-table information related to a specific VLAN. Valid range is 1 to 4094. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following is an example of the **show mac address-table interface eth 0/1** command displaying MAC address-table entries specifically on Ethernet 0/1:

```
>enable
#show mac address-table interface ethernet 0/1
Mac Address Table
Vlan    Mac Address          Type        Ports
2       00:90:2b:7d:30:00    DYNAMIC     eth 0/1
2       00:a0:c8:05:00:ac    DYNAMIC     eth 0/1
2       00:a0:c8:05:00:ad    DYNAMIC     eth 0/1
2       00:a0:c8:05:00:c2    DYNAMIC     eth 0/1
2       00:a0:c8:05:01:6e    DYNAMIC     eth 0/1
2       00:a0:c8:09:95:6b    DYNAMIC     eth 0/1
2       00:a0:c8:0a:2d:7c    DYNAMIC     eth 0/1
Total Mac Addresses for this criterion: 10
```

# show mac address-table static

Use the **show mac address-table static** command to display all static medium access control (MAC) addresses known by AOS. Variations of this command include the following:

**show mac address-table static**
**show mac address-table static address** <*mac address*>
**show mac address-table static address** <*mac address*> **interface** <*interface*>
**show mac address-table static address** <*mac address*> **interface** <*interface*> **vlan** <*vlan id*>
**show mac address-table static address** <*mac address*> **vlan** <*vlan id*>
**show mac address-table static interface** <*interface*>
**show mac address-table static interface** <*interface*> **vlan** <*vlan id*>
**show mac address-table static vlan** <*vlan id*>

## Syntax Description

| | |
|---|---|
| **address** <*mac address*> | Optional. Specifies a valid 48-bit MAC address. MAC addresses should be expressed in following format xx:xx:xx:xx:xx:xx (for example, **00:A0:C8:00:00:01**). |
| **interface** <*interface*> | Optional. Shows information for a specific interface. Specify an interface in the format <*interface type [slot/port* \| *slot/port.sub-interface id* \| *interface id* \| *interface id.sub-interface id]*>. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **show mac address-table static interface ?** for a list of valid interfaces. |
| **vlan** <*vlan id*> | Optional. Shows address-table information related to a specific VLAN. Valid range is 1 to 4094. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following is a sample from the **show mac address-table static** command:

```
>enable
#show mac address-table static
Mac Address Table
------------------------------
Vlan    Mac Address          Type        Ports
------  -------------------  --------    --------
 1      00:a0:c8:00:88:40    STATIC      CPU
Total Mac Addresses for this criterion: 1
```

# show media-gateway

Use the **show media-gateway** command to show cumulative totals for all Realtime Transfer Protocol (RTP) channels. Variations of this command include:

**show media-gateway**
**show media-gateway channel**
**show media-gateway channel** <*slot/dsp.channel*>
**show media-gateway info**
**show media-gateway session**
**show media-gateway session** <*slot/dsp.channel*>
**show media-gateway summary**
**show media-gateway summary active**

## Syntax Description

| | |
|---|---|
| <*slot/dsp.channel*> | Optional. Specifies the ID of the media-gateway channel to be displayed. |
| **channel** | Optional. Shows cumulative totals for individual RTP channels. |
| **info** | Optional. Shows media-gateway information. |
| **session** | Optional. Shows current RTP sessions. |
| **summary** | Optional. Shows summary of last active and current RTP sessions. |
| **active** | Optional. Shows summary of currently active RTP sessions. |

## Default Values

No default necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example shows sample output from the show **media-gateway** command:

```
>enable
#show media-gateway
Media-Gateway
 1 slots, 1 DSPs, 24 channels
 0 total sessions, 0 active sessions, 00:00:00 total session duration
 0 total rx packets, 0 total rx bytes
 0 total lost packets, 0 total unknown packets
 0 total tx packets, 0 total tx bytes
 0 highest max depth
 0 total discards, 0 total overflows, 0 total underflows
 0 total out-of-orders
 Last clearing of counters: 9:46 PM Thu, Jan 1, 1970
#
```

# show memory

Use the **show memory** command to display statistics regarding memory including memory allocation and buffer use statistics. Shows how memory is in use (broken down by memory size) and how much memory is free. Variations of this command include:

**show memory heap**
**show memory heap realtime**
**show memory uncached-heap**


*Using the **realtime** argument for this command can adversely affect the performance or your unit.*

## Syntax Description

| | |
|---|---|
| **heap** | Shows how much memory is in use (broken down by memory block size) and how much memory is free. |
| **realtime** | Optional. Displays full-screen output in real time. Refer to the *Functional Notes* below for more information. |
| **uncached-heap** | Shows how much memory has been set aside to be used without memory caching, how much memory is being used and how much memory is free. (Valid only on NetVanta 300, 1000, and 1000R Series Units.) |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |
| Release 10.1 | The real time display option was introduced. |

## Functional Notes

Use the **realtime** argument for this command to display full-screen output in real time. Information is continuously updated on the console until you either freeze the data (by pressing the **F** key) or exit **realtime** mode (by pressing **Ctrl-C**). If there is not enough room on the screen for all available data, the information will truncate at the bottom of the screen. In order to maximize the amount of data displayed, increase the terminal length (using the **terminal length** command; refer to *terminal length <number>* on page 410).

## Usage Examples

The following is a sample output from the **show memory heap** command:

>**enable**
**#show memory heap**
Memory Heap:
  HeapFree: 2935792
  HeapSize: 8522736
Block Managers:

| Mgr | Size | Used | Free | Max-Used |
|---|---|---|---|---|
| 0 | 0 | 58 | 0 | 58 |
| 1 | 16 | 1263 | 10 | 1273 |
| 2 | 48 | 1225 | 2 | 1227 |
| 3 | 112 | 432 | 2 | 434 |
| 4 | 240 | 140 | 3 | 143 |
| 5 | 496 | 72 | 2 | 74 |
| 6 | 1008 | 76 | 1 | 26 |
| 7 | 2032 | 25 | 1 | 26 |
| 8 | 4080 | 2 | 1 | 3 |
| 9 | 8176 | 31 | 1 | 32 |
| 10 | 16368 | 8 | 0 | 8 |
| 11 | 32752 | 5 | 1 | 6 |
| 12 | 65520 | 3 | 0 | 30 |
| 13 | 131056 | 0 | 0 | 0 |

# show modules

The **show modules** command displays information on the current system setup. Variations of this command include:

**show modules**
**show modules verbose**

## Syntax Description

| | |
|---|---|
| **verbose** | Optional. Enables detailed messaging. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Usage Examples

The following example displays the modules installed in the unit.

```
>enable
#show modules
```

| Slot | Ports | Type | Serial # | Part # | H/W Rev |
|---|---|---|---|---|---|
| 0 | 3 | Netvanta 5305 | ********** | 1200990L1 | A |
| 1 | 1 | T3 Module | ********** | 1200832L1 | A |
| 2 | - | Empty | ---------- | ---------- | ---------- |
| 3 | - | Empty | ---------- | ---------- | ---------- |
| 4 | - | Empty | ---------- | ---------- | ---------- |
| 5 | - | Empty | ---------- | ---------- | ---------- |
| 6 | - | Empty | ---------- | ---------- | ---------- |
| 7 | - | Empty | ---------- | ---------- | ---------- |

# show monitor session

Use the **show monitor session** command to display information regarding a specified monitor session or to display this information for all sessions. Variations of this command include:

**show monitor session** <*number*>
**show monitor session all**

## Syntax Description

| | |
|---|---|
| <*number*> | Displays information for a single specific monitor session. |
| **all** | Displays all sessions. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following is a sample from the **show monitor session** command:

```
>enable
#show monitor session 1
Monitor Session 1
-----------------
Source Ports:
   RX Only:   None
   TX Only:   None
   Both:      eth 0/2, eth 0/3
Destination Port: eth 0/6
```

# show output-startup

Use the **show output-startup** command to display startup configuration output line-by-line. This output can be copied into a text file and then used as a configuration editing tool.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Usage Examples

The following is a sample output from the **show output-startup** command:

```
>enable
#show output-startup
!
#!
#hostname "UNIT_2"
UNIT_2#no enable password
UNIT_2#!
UNIT_2#ip subnet-zero
UNIT_2#ip classless
UNIT_2#ip routing
UNIT_2#!
UNIT_2#event-history on
UNIT_2#no logging forwarding
UNIT_2#logging forwarding priority-level info
UNIT_2#no logging email
etc....
```

# show port-auth

Use the **show port-auth** command to view port authentication information. Variations of this command include:

**show port-auth**
**show port-auth detailed**
**show port-auth detailed interface** <*interface*>
**show port-auth interface** <*interface*>
**show port-auth statistics**
**show port-auth statistics interface** <*interface*>
**show port-auth summary**
**show port-auth summary interface** <*interface*>

## Syntax Description

| | |
|---|---|
| **detailed** | Optional. Displays detailed port authentication information. |
| **interface** <*interface*> | Optional. Displays port authentication information for the specified interface. Specify an interface in the format <*interface type [slot/port | slot/port.sub-interface id | interface id | interface id.sub-interface id]*>. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **show port-auth interface ?** for a list of valid interfaces. |
| **statistics** | Optional. Displays port authentication statistics. |
| **summary** | Optional. Displays a summary of port authentication settings. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example displays the port authentication information:

```
>enable
#show port-auth
Global Port-Authentication Parameters:
   re-authentication enabled: False
   reauth-period: 3600
   quiet-period: 60
   tx-period: 30
   supp-timeout: 30
   server-timeout: 30
   reauth-max:2
```

Port-Authentication Port Summary:

| Interface | Status | Type | Mode | Authorized |
|---|---|---|---|---|
| eth 0/1 | disabled | port-based | n/a | n/a |
| eth 0/2 | disabled | port-based | n/a | n/a |
| eth 0/3 | disabled | port-based | n/a | n/a |
| eth 0/4 | disabled | port-based | n/a | n/a |
| eth 0/5 | disabled | port-based | n/a | n/a |
| eth 0/6 | disabled | port-based | n/a | n/a |
| eth 0/7 | disabled | port-based | n/a | n/a |
| eth 0/8 | disabled | port-based | n/a | n/a |
| eth 0/9 | disabled | port-based | n/a | n/a |
| eth 0/10 | disabled | port-based | n/a | n/a |
| eth 0/11 | disabled | port-based | n/a | n/a |
| eth 0/12 | disabled | port-based | n/a | n/a |
| eth 0/13 | disabled | port-based | n/a | n/a |
| eth 0/14 | disabled | port-based | n/a | n/a |
| eth 0/15 | disabled | port-based | n/a | n/a |
| eth 0/16 | disabled | port-based | n/a | n/a |
| eth 0/17 | disabled | port-based | n/a | n/a |
| eth 0/18 | disabled | port-based | n/a | n/a |
| eth 0/19 | disabled | port-based | n/a | n/a |
| eth 0/20 | disabled | port-based | n/a | n/a |
| eth 0/21 | disabled | port-based | n/a | n/a |
| eth 0/22 | disabled | port-based | n/a | n/a |
| eth 0/23 | disabled | port-based | n/a | n/a |
| eth 0/24 | disabled | port-based | n/a | n/a |

Port Authentication Port Details:

Port-Authentication is disabled on eth 0/1

Port-Authentication is disabled on eth 0/2

# show port-security

Use the **show port-security** command to display port security information. Variations of this command include:

**show port-security**
**show port-security address**
**show port-security interface** <*interface*>
**show port-security interface** <*interface*> **address**
**show port-security port-expiration**
**show port-security port-expiration detailed**

## Syntax Description

| | |
|---|---|
| **address** | Displays a list of secure medium access control (MAC) addresses for all interfaces currently configured for port security. |
| **interface** <*interface*> | Filters the output to include only information for the specified interface. Specify an interface in the format *<interface type [slot/port \| slot/port.sub-interface id \| interface id \| interface id.sub-interface id]>*. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **show port-security interface ?** for a complete list of valid interfaces. |
| **port-expiration** | Displays the ports currently participating in port expiration and the amount of time left until the port is shutdown. |
| **detailed** | Displays information for all interfaces, even if not configured for port expiration. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Usage Examples

The following displays all secure MAC addresses related to the Ethernet 0/1 interface:

```
>enable
#show port-security interface eth 0/1 address
VLAN   Mac Address         Type of Entry     Interface   Remaining Time
-------------------------------------------------------------------------------------------------------------
1      00:a0:c8:0a:c6:4a   Dynamic-Secure    eth 0/1     --
1      00:a0:c8:0a:c6:4b   Dynamic-Secure    eth 0/1     --
-------------------------------------------------------------------------------------------------------------
Dynamic Address Count: 2
Static Address Count: 0
Sticky Address Count: 0
Total Address Count: 2
```

# show power inline

Use the **show power inline** command to display power information (in watts) for devices connected to power over Ethernet (PoE) interfaces. The command also displays the PoE interfaces that can be powered, whether the interfaces are powered or not, and the IEEE class for the device(s) connected to the PoE interfaces. Variations of this command include:

**show power inline**
**show power inline** <*slot/port*>
**show power inline** <*slot/port*> **realtime**


*Using the **realtime** argument for this command can adversely affect the performance or your unit.*

## Syntax Description

| | |
|---|---|
| <*slot/port*> | Optional. Specifies the slot/port of a PoE interface. If specified, the command only displays information related to that interface. |
| **realtime** | Optional. Displays full-screen output in real time. Refer to the *Functional Notes* below for more information. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |
| Release 11.1 | The real time display option was added. |

## Functional Notes

Use the **realtime** argument for this command to display full-screen output in real time. Information is continuously updated on the console until you either freeze the data (by pressing the **F** key) or exit **realtime** mode (by pressing **Ctrl-C**). If there is not enough room on the screen for all available data, the information will truncate at the bottom of the screen. In order to maximize the amount of data displayed, increase the terminal length (using the **terminal length** command; refer to *terminal length <number>* on page 410).

## Usage Examples

The following example displays power information for all PoE interfaces:

```
>enable
#show power inline
Interface     Admin     Oper     Power (watts)     Class
eth 0/1       auto      off      n/a               n/a
eth 0/2       auto      off      n/a               n/a etc. . .
```

# show power-supply

The **show power-supply** command displays the power supply status.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Usage Examples

The following example displays the power supply status:

```
>enable
#show power supply
Power supply 1 is OK.
Power supply 2 is not present.
```

# show pppoe

Use the **show pppoe** command to display all point-to-point over Ethernet (PPPoE) settings and associated parameters.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Usage Examples

The following example enters the Enable mode and uses the **show** command to display PPPoE information:

```
>enable
#show pppoe
ppp 1
 Outgoing Interface: eth 0/1
 Outgoing Interface MAC Address: 00:A0:C8:00:85:20
 Access-Concentrator Name Requested: FIRST VALID
 Access-Concentrator Name Received: 13021109813703-LRVLGAOS90W_IFITL
 Access-Concentrator MAC Address: 00:10:67:00:1D:B8
 Session Id: 64508
 Service Name Requested: ANY
 Service Name Available:
 PPPoE Client State: Bound (3)
 Redial retries: unlimited
 Redial delay: 10 seconds
Backup enabled all day on the following days:
    Sunday Monday Tuesday Wednesday Thursday Friday Saturday
 Backup phone number list:
     Number      Call Type     min/max DS0s     Backup I/F
     5551212     analog        1/1              ppp 2
```

# show probe

Use the **show probe** command to display probe configuration and statistics. Refer to *Network Monitor Probe Command Set* on page 1682 for information on configuring probe objects. Variations of this command include the following:

**show probe**
**show probe** <*name*>
**show probe** <*name*> **realtime**

## Syntax Description

| | |
|---|---|
| <*name*> | Optional. Displays configuration and statistics for a specific probe. |
| **realtime** | Optional. Displays full-screen output in real time. Refer to the *Functional Notes* below for more information. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Functional Notes

A probe must be created first using the **probe** command. Issuing the shutdown command at the **probe** configuration prompt will disable a probe, causing it to cease traffic generation. While a probe is shutdown, it will not fail.

Use the **realtime** argument for this command to display full-screen output in real time. Information is continuously updated on the console until you either freeze the data (by pressing the **F** key) or exit **realtime** mode (by pressing **Ctrl-C**). If there is not enough room on the screen for all available data, the information will truncate at the bottom of the screen. In order to maximize the amount of data displayed, increase the terminal length (using the **terminal length** command; refer to *terminal length <number>* on page 410).

## Usage Examples

The following is a sample output of the **show probe probe_A** command:

```
>enable
#show probe probe_A
Current State: PASS    Admin. Status: DOWN
    Type: ICMP Echo Period: 30 sec Timeout: 500 msec
    Hostname: www.adtran.com
    Tracked by: track_1
    Tests Run: 121 Failed: 0
    Time in current state: 25 days 2 hours, 34 minutes, 32 seconds
```

# show processes

Use the **show processes** command to display process statistic information. Variations of this command include:

**show processes cpu**
**show processes cpu realtime**
**show processes history**
**show processes queue**

*Using the **realtime** argument for this command can adversely affect the performance or your unit.*

## Syntax Description

| | |
|---|---|
| **cpu** | Displays informations about processes that are currently active. |
| **realtime** | Optional. Displays full-screen CPU output in real time. Refer to the *Functional Notes* below for more information. |
| **history** | Displays the process switch history. |
| **queue** | Displays process queue utilization. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |
| Release 10.1 | New option was introduced. |

## Functional Notes

Use the **realtime** argument for this command to display full-screen output in real time. Information is continuously updated on the console until you either freeze the data (by pressing the **F** key) or exit **realtime** mode (by pressing **Ctrl-C**). If there is not enough room on the screen for all available data, the information will truncate at the bottom of the screen. In order to maximize the amount of data displayed, increase the terminal length (using the **terminal length** command; refer to *terminal length <number>* on page 410).

## Usage Examples

The following is a sample output from the **show processes cpu** command:

>**enable**
**#show processes cpu**
processes cpu
System load: 7.07%  Min: 0.00%  Max 85.89%
Context switch load: 0.21%

| Task | Task | Invoked | Exec | Time | Runtime | Load % |
|---|---|---|---|---|---|---|
| D | Name | PRI STAT | (count) | (usec) | (usec) | (1sec) |
| 0 | Idle | 0 W | 129689 | 1971 | 927923 | 92.79 |
| 1 | FrontPanel | 249 W | 9658 | 165 | 3202 | 0.32 |
| 3 | Stack Usage | 11 W | 485 | 305 | 325 | 0.03 |
| 4 | Q Test 1 | 10 W | 50 | 4 | 0 | 0.00 |
| 5 | Q Test 2 | 11 W | 50 | 6 | 0 | 0.00 |

....etc.

# show qos

Use the **show qos** command to display information regarding quality of service (QoS) and cost of service (CoS) settings. Variations of this command include:

**show qos cos-map**
**show qos dscp-cos**
**show qos interface** <*interface*>
**show qos queuing**

## Syntax Description

| | |
|---|---|
| **cos-map** | Displays the CoS priority-to-queue map. The map outlines which CoS priority is associated with which queue. |
| **dscp-cos** | Displays the Differentiated Services Code Point (DSCP) to CoS map settings. |
| **interface** <*interface*> | Displays the QoS settings on a specific interface. Specify an interface in the format <*interface type [slot/port | slot/port.sub-interface id | interface id | interface id.sub-interface id]*>. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **show qos interface ?** for a complete list of valid interfaces. |
| **queuing** | Displays the type of queuing being used. If weighted round robin (WRR) queuing is enabled, the command also displays the weight of each queue. |

## Default Values

No defaults necessary for this command.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |
| Release 7.1 | Command was expanded to include the **dscp-cos** option. |

## Usage Examples

The following is sample output from the **show qos cos-map** command:

```
>enable
#show qos cos-map
CoS Priority: 0 1 2 3 4 5 6 7
Priority Queue: 1 1 2 2 3 3 4 4
```

The following is sample output from the **show qos interface** command for Ethernet 0/8 interface:

```
>enable
#show qos interface ethernet 0/8
 Ethernet 0/8
 trust state: trusted
 default CoS: 0
```

The following is sample output from the **show qos queuing** command with WRR queuing enabled:

```
>enable
#show qos queuing
Queue-type: wrr
Expedite queue: disabled
wrr weights:
 qid - weight
 1 - 12
 2 - 45
 3 - 55
 4 - 65
```

# show qos map

The **show qos map** command outputs information about the quality of service (QoS) map. This information differs based on how a particular map entry is defined. Variations of this command include the following:

**show qos map**
**show qos map** <*name*>
**show qos map** <*name*> <*number*>
**show qos map interface** <*interface*>

## Syntax Description

| | |
|---|---|
| <*name*> | Optional. Specifies the name of a defined QoS map. |
| <*number*> | Optional. Specifies one of the map's defined sequence numbers. |
| **interface** <*interface*> | Optional. Displays the QoS map information for a specific interface. Specify an interface in the format <*interface type [slot/port | slot/port.sub-interface id | interface id | interface id.sub-interface id]*>. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **show qos map interface ?** command for a complete list of interfaces. |

## Default Values

No defaults necessary for this command.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |
| Release 9.1 | Command was expanded to include HDLC interface. |
| Release 11.1 | Demand interface was added. |

## Usage Examples

The following example shows all Qos Maps and all entries in those maps:

```
>enable
#show qos map
qos map priority
  map entry 10
    match IP packets with a precedence value of 6
    priority bandwidth: 400 (kilobits/sec) burst: default
    packets matched by map: 125520
  map entry 20
    match ACL icmp
    packets matched by map: 99
  map entry 30
    match RTP packets on even destination ports between 16000 and 17000
```

```
    packets matched by map: 0
  map entry 50
    match ACL tcp
    packets matched by map: 4326
  map entry 60
  match IP packets with a dscp value of 2
  set dscp value to 6
  packets matched by map: 0
  map entry 70
  match NetBEUI frames being bridged by the router
  priority bandwidth: 150 (kilobits/sec) burst: default
  packets matched by map: 0
  qos map tcp_map
  map entry 10
    match ACL tcp
    priority bandwidth: 10 (kilobits/sec) burst: default
    set precedence value to 5
    packets matched by map: 0
  map entry 20
    match IP packets with a precedence value of 3
    priority bandwidth: 50 (kilobits/sec) burst: default
    packets matched by map: 0
```

The following example shows the Qos Map named **priority** and all entries in that map:

```
>enable
#show qos map priority
  qos map priority
  map entry 10
    match IP packets with a precedence value of 6
    priority bandwidth: 400 (kilobits/sec) burst: default
    packets matched by map: 125520
  map entry 20
    match ACL icmp
    packets matched by map: 99
  map entry 30
    match RTP packets on even destination ports between 16000 and 17000
    packets matched by map: 0
  map entry 50
    match ACL tcp
    packets matched by map: 4326
  map entry 60
    match IP packets with a dscp value of 2
    set dscp value to 6
    packets matched by map: 0
  map entry 70
```

```
    match NetBEUI frames being bridged by the router
    priority bandwidth: 150 (kilobits/sec) burst: default
    packets matched by map: 0
```

The following example shows only QoS map named **priority** with the sequence number **10**:

```
>enable
#show qos map priority 10
   qos map priority
   map entry 10
    match IP packets with a precedence value of 6
    priority bandwidth: 400 (kilobits/sec) burst: default
    packets matched by map: 125520
```

The following examples show Qos Map interface stats associated with the map defined for an interface:

```
>enable
#show qos map interface frame-relay 1
fr 1
qos-policy out: priority
   map entry 10
    match IP packets with a precedence value of 6
    budget 145/10000 bytes (current/max)
    priority bandwidth: 400 (kilobits/sec)
    packets matched on interface: 27289
    packets dropped: 98231
   map entry 20
   not configured for rate limiting

   map entry 30
   not configured for rate limiting

   map entry 50
   not configured for rate limiting

   map entry 60
   not configured for rate limiting

   map entry 70
    match NetBEUI frames being bridged by the router
    budget 3750/3750 bytes (current/max)
    priority bandwidth: 150 (kilobits/sec)
    packets matched on interface: 0
    packets dropped: 0
```

# show queue <*interface*>

Use the **show queue** command to display conversation information associated with an interface queue. This command shows summary and per-conversation information.

## Syntax Description

| | |
|---|---|
| <*interface*> | Displays the queueing information for the specified interface. Specify an interface in the format <*interface type [slot/port | slot/port.sub-interface id | interface id | interface id.sub-interface id]*>. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type the **show queue ?** command to display a list of valid interfaces. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |
| Release 9.1 | Command was expanded to include HDLC interface. |
| Release 11.1 | Demand interface was added. |

## Usage Examples

The following is a sample output from the **show queue** command:

```
>enable
#show queue fr 1
Queueing method: weighted fair
 Output queue: 18/25/200/64/1027 (size/highest/max total/threshold/drops)
    Conversations 2/4/256 (active/max active/max total)

 (depth/weight/highest/discards) 12/256/33/0
 Conversation 10, linktype: ip, length: 67
 source: 10.100.23.11, destination: 10.200.2.125, id: 0x0000, ttl: 47,
 TOS: 0 prot: 17 (udp), source port 99, destination port 99

 (depth/weight/highest/discards) 6/256/25/0
 Conversation 23, linktype: ip, length: 258
 source: 10.100.23.11, destination: 10.200.2.125, id: 0x0000, ttl: 47,
 TOS: 0 prot: 6 (tcp), source port 16, destination port 16
```

# show queuing

Use the **show queuing** command to display information associated with configured queuing methods. Variations of this command include:

**show queuing**
**show queuing fair**

## Syntax Description

| | |
|---|---|
| **fair** | Optional. Displays only information on the weighted fair queuing configuration. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following is a sample output from the **show queuing** command:

```
>enable
#show queuing
  Interface        Discard threshold    Conversation subqueues
  fr 1             64                   256
  fr 2             64                   256
  ppp 1            64                   256
```

# show radius statistics

Use the **show radius statistics** command to display various statistics from the RADIUS subsystem. These statistics include number of packets sent, number of invalid responses, number of timeouts, average packet delay, and maximum packet delay. Statistics are shown for both authentication and accounting packets.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following is an example output using the **show radius statistics** command:

```
>enable
#show radius statistics
                              Auth.       Acct.
Number of packets sent:       3           0
Number of invalid responses:  0           0
Number of timeouts:           0           0
Average delay:                2 ms        0 ms
Maximum delay:                3 ms        0 ms
```

# show route-map

Use the **show route-map** command to display any route-maps that have been configured in the router. It displays any match and set clauses associated with the route-map, as well as the number of incoming routes that have matched each route-map. Route-maps can be used for BGP and PBR. Variations of this command include:

**show route-map**
**show route-map** <*name*>

## Syntax Description

| | |
|---|---|
| <*name*> | Optional. Displays only the route map matching the specified name. |

## Default Values

By default, this command displays all defined route-maps.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

In the example below, all route-maps in the router are displayed.

```
>enable
#show route-map
route-map RouteMap1, permit, sequence 10
 Match clauses:
   community (community-list filter): CommList1
 Set clauses:
   local-preference 250
 BGP Filtering matches: 75 routes
 Policy routing matches: 0 packets 0 bytes
route-map RouteMap1, permit, sequence 20
 Match clauses:
   community (community-list filter): CommList2
 Set clauses:
   local-preference 350
 BGP Filtering matches: 87 routes
 Policy routing matches: 0 packets 0 bytes
route-map RouteMap2, permit, sequence 10
 Match clauses:
   ip address (access-lists): Acl1
 Set clauses:
   metric 100
 BGP Filtering matches: 10 routes
 Policy routing matches: 0 packets 0 bytes
```

```
route-map RouteMap2, permit, sequence 20
 Match clauses:
   ip address (access-lists): Acl2
 Set clauses:
   metric 200
 BGP Filtering matches: 12 routes
 Policy routing matches: 0 packets 0 bytes
route-map RouteMap3, permit, sequence 10
 Match clauses:
   length 150 200
 Set clauses:
   ip next-hop: 10.10.11.254
 BGP Filtering matches: 0 routes
 Policy routing matches: 0 packets 0 bytes
route-map RouteMap3, permit, sequence 20
 Match clauses:
   ip address (access-lists): Acl3
 Set clauses:
   ip next-hop: 10.10.11.14
 BGP Filtering matches: 0 routes
 Policy routing matches: 144 packets 15190 bytes
```

In the example below, only **RouteMap2** is displayed.

```
>enable
#show route-map RouteMap2
route-map RouteMap2, permit, sequence 10
 Match clauses:
   ip address (access-lists): Acl1
 Set clauses:
   metric 100
 BGP Filtering matches: 10 routes
 Policy routing matches: 0 packets 0 bytes
route-map RouteMap2, permit, sequence 20
 Match clauses:
   ip address (access-lists): Acl2
 Set clauses:
   metric 200
 BGP Filtering matches: 12 routes
 Policy routing matches: 0 packets 0 bytes
```

# show rtp resources

Use the **show rtp resources** command to display Realtime Transfer Protocol (RTP) resource information.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following is sample output for the **show rtp resources** command:

```
>enable
#show rtp resources
DSP      Channel    Type     Port      Status
-------------------------------------------------------------------------
 0/1       1         RTP      N/A       Available
 0/1       2         RTP       N/A      Available
 0/1       3         RTP      N/A       Available
 0/1       4         RTP      N/A       Available
 0/1       5         RTP      N/A       Available
 0/1       6         RTP      N/A       Available
 0/1       7         RTP      N/A       Available
 0/1       8         RTP      N/A       Available
 0/1       9         RTP      N/A       Available
 0/1       10        RTP      N/A       Available
 0/1       11        RTP      N/A       Available
 0/1       12        RTP      N/A       Available
 0/1       13        RTP      N/A       Available
 0/1       14        RTP      N/A       Available
 0/1       15        RTP      N/A       Available
 0/1       16        RTP      N/A       Available
 0/1       17        RTP      N/A       Available
 0/1       18        RTP      N/A       Available
 0/1       19        RTP      N/A       Available
 0/1       20        RTP      N/A       Available
```

## show running-config

Use the **show running-config** command to display a text print of all the non-default parameters contained in the current running configuration file. Specific portions of the running configuration may be displayed, based on the command entered. Variations of this command include the following:

**show running-config**
**show running-config access-lists**
**show running-config access-lists verbose**
**show running-config checksum**
**show running-config interface** <*interface*>
**show running-config interface** <*interface*> **verbose**
**show running-config ip-crypto**
**show running-config ip-crypto verbose**
**show running-config ip rtp**
**show running-config ip rtp verbose**
**show running-config ip sdp**
**show running-config ip sdp verbose**
**show running-config ip sip**
**show running-config ip sip verbose**
**show running-config policy-class**
**show running-config policy-class verbose**
**show running-config probe**
**show running-config probe verbose**
**show running-config qos-map**
**show running-config qos-map verbose**
**show running-config router pim-sparse**
**show running-config router pim-sparse verbose**
**show running-config track**
**show running-config track verbose**
**show running-config verbose**

### Syntax Description

| | |
|---|---|
| **access-lists** | Displays the current running configuration for all configured IP access control lists. |
| **checksum** | Displays the encrypted Message Digest 5 (MD5) version of the running configuration. |
| **interface** <*interface type*> | Displays the current running configuration for a particular interface. Specifies an interface in the format <*interface type [slot/port | slot/port.sub-interface id | interface id | interface id.sub-interface id]*>. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **show running-config interface?** for a complete list of valid interfaces. |
| **ip crypto** | Displays the current running configuration for all IPSec VPN settings. |

| | |
|---|---|
| **policy-class** | Displays the current running configuration for all configured policy classes. |
| **ip rtp** | Displays the current running configuration for all IP Real-time Transport Protocol (RTP) parameters. |
| **ip sdp** | Displays the current running configuration for all Session Description Protocol (SDP) parameters. |
| **policy-class** | Displays the current running configuration for all configured policy classes. |
| **probe** | Displays the current configuration for all running probes. |
| **qos-map** | Displays the current running configuration for all configured QoS maps. |
| **router pim-sparse** | Optional. Displays the current global PIM-SM configuration. |
| **track** | Displays the current running configuration for all tracks. |
| **verbose** | Optional. Displays the entire running configuration to the terminal screen (versus only the non-default values). |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 9.1 | Command was expanded to include HDLC and tunnel interfaces. |
| Release 11.1 | Demand, FXO, and serial interfaces were added. IP crypto and router pim-sparse key words were added. |
| Release 13.1 | Command was expanded to include ip rtp, ip sdp, probe and track subcommands+. |

## Usage Examples

The following is a sample output from the **show running-config** command:

```
>enable
#show running-config
Building configuration...
!
no enable password
!
ip subnet-zero
ip classless
ip routing
!
event-history on
no logging forwarding
logging forwarding priority-level info
no logging email
!
ip policy-timeout tcp all-ports 600
ip policy-timeout udp all-ports 60
ip policy-timeout icmp 60
!
interface eth 0/1.....
```

## show running-config voice

Use the **show running-config voice** command to show running voice configurations.Variations of this command include the following:

**show running-config voice**
**show running-config voice ani**
**show running-config voice ani verbose**
**show running-config voice autoattendant**
**show running-config voice autoattendant verbose**
**show running-config voice class-of-service**
**show running-config voice class-of-service verbose**
**show running-config voice class-of-service** <*name*>
**show running-config voice class-of-service** <*name*> **verbose**
**show running-config voice grouped-trunk**
**show running-config voice grouped-trunk verbose**
**show running-config voice grouped-trunk** <*name*>
**show running-config voice grouped-trunk** <*name*> **verbose**
**show running-config voice mail**
**show running-config voice mail verbose**
**show running-config voice operator-group**
**show running-config voice operator-group verbose**
**show running-config voice ring-group**
**show running-config voice ring-group** <*name*>
**show running-config voice speed-dial**
**show running-config voice speed-dial verbose**
**show running-config voice trunk**
**show running-config voice trunk verbose**
**show running-config voice trunk** <*Txx*>
**show running-config voice trunk** <*Txx*> **verbose**
**show running-config voice user**
**show running-config voice user verbose**
**show running-config voice user** <*number*>
**show running-config voice user** <*number*> **verbose**
**show running-config voice verbose**

### Syntax Description

| | |
|---|---|
| **ani** | Displays ANI substitution configurations. |
| **autoattendant** | Displays autoattendant configuration. |
| **class-of-service** <*name*> | Displays voice class of service (CoS) configurations for the specified rule set. |
| **grouped-trunk** <*name*> | Displays voice trunk group configurations for the specified trunk. |
| **mail** | Displays running voice mail configuration. |
| **operator-group** | Displays the running Operator Group configuration. |
| **ring-group** <*name*> | Displays the running ring group configurations for the specified ring group. |

| | |
|---|---|
| **speed-dial** | Displays all entries for speed-dial entries. |
| **trunk** <*Txx*> | Displays voice trunk configurations for the specified trunk.Use the trunk's 2-digit identifier following T (for example, T99). |
| **user** <*number*> | Displays voice user configurations for the specified user. |
| **verbose** | Displays detailed voice running configurations. |

## Default Values

No default necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |
| Release 11.1 | Speed-dial option was added. |
| Release 13.1 | Command was expanded. |

## Usage Examples

The following example shows sample output from the **show running-config voice** command:

```
>enable
#show voice running-config voice
Building configuration...
!
voice hold-reminder 15
voice flashhook mode interpreted
!
voice dial-plan 1 local 8000
!
voice class-of-service set1
 billing-codes
!
voice class-of-service set2
!
voice class-of-service “set 1"
!
voice codec-list trunk
 default
 codec g711ulaw
 codec g729
!
voice codec-list “list 1"
!
voice codec-list list1
!
voice trunk T99 type t1-rbs supervision wink role network
!
```

```
voice trunk T01 type sip
!
voice trunk T07 type t1-rbs supervision wink role network
!
voice trunk T02 type t1-rbs supervision wink role network
!
voice trunk T03 type t1-rbs supervision wink role network
!
voice trunk T12 type sip
!
voice grouped-trunk TEST
 no description
 reject 900XXXXXXX
!
voice grouped-trunk TESTGROUP
 no description
```

# show sip location

Use the **show sip location** command to display Session Initiation Protocol (SIP) statistical and registration information. Variations of this command include:

**show sip location**
**show sip location dynamic**
**show sip location static**

## Syntax Description

| | |
|---|---|
| **dynamic** | Optional. Displays SIP location database dynamic entries. |
| **static** | Optional. Displays SIP location database static entries. |

## Default Values

No default necessary for this command.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |
| Release 12.1 | Command output was updated. |

## Usage Examples

The following example shows sample output from the **show sip location static** command:

```
>enable
#show sip location static
User          IP Address      Port Expires Source
-------------------------------------------------------------
9001          1.1.1.2         5060     52 Registrar
9002          10.10.10.2      5060   3336 Registrar
```

# show sip

Use the **show sip** command to display Session Initiation Protocol (SIP) statistical and registration information. Variations of this command include:

**show sip resources**
**show sip statistics**
**show sip user-registration**

## Syntax Description

| | |
|---|---|
| **resources** | Displays SIP server resource information. |
| **statistics** | Displays SIP server statistic information. |
| **user-registration** | Displays local SIP server registration information. |

## Default Values

No default necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |
| Release 11.1 | Resources, statistics, and user-registration options were added. |

## Usage Examples

The following example shows sample output from the **show sip statistics** command:

>**enable**
**#show sip statistics**

Invites transmitted: 36
Invites received: 26

Invite Retransmits transmitted: 11
Invite Retransmits received: 0

Non-Invites transmitted: 1869
Non-Invites received: 1911

Non-Invite Retransmits transmitted: 12
Non-Invite Retransmits received: 41

Responses transmitted: 1982
Responses received: 3535

Response Retransmits transmitted: 45
Response Retransmits received: 0

The following example shows sample output from the **show sip user-registration** command:

>**enable**
**#show sip user-registration**

| EXT. | TYPE | IP ADDRESS | PORT | EXP |
|---|---|---|---|---|
| 9001 | SIP - Generic | 1.1.1.2 | 5060 | 25 |
| 9002 | SIP - Generic | 10.10.10.2 | 5060 | 3419 |

Total phones registered: 2

# show sip trunk-registration

Use the **show sip trunk-registration** command to display Session Initiation Protocol (SIP) statistical and registration information. Variations of this command include the following:

**show sip trunk-registration**
**show sip trunk-registration realtime**
**show sip trunk-registration** <*Txx*>
**show sip trunk-registration** <*Txx*> **realtime**
**show sip trunk-registration** <*Txx*> <*name*>
**show sip trunk-registration** <*Txx*> <*name*>> **realtime**

*Using the **realtime** argument for this command can adversely affect the performance or your unit.*

## Syntax Description

| | |
|---|---|
| <*Txx*> | Optional. Specifies the trunk identity; where *xx* is the trunk's two-digit identifier, e.g., T01. |
| <*name*> | Optional. Specifies the name associated with the trunk. |
| **realtime** | Optional. Displays local SIP client registration information in real time. Refer to the *Functional Notes* below for more information. |

## Default Values

No default necessary for this command.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Functional Notes

Use the **realtime** argument for this command to display full-screen output in real time. Information is continuously updated on the console until you either freeze the data (by pressing the **F** key) or exit **realtime** mode (by pressing **Ctrl-C**). If there is not enough room on the screen for all available data, the information will truncate at the bottom of the screen. In order to maximize the amount of data displayed, increase the terminal length (using the **terminal length** command; refer to *terminal length <number>* on page 410).

## Usage Examples

The following example shows sample output from the **show sip trunk-registration** command:

>**enable**
**#show sip trunk-registration**

| Ext | Register | Expire | Grant | Success | Redirect | Challenge | Failed | Timeout |
|---|---|---|---|---|---|---|---|---|
| 4433 | NO | 0 | 0 | 0 | 0 | 0 | 0 | # |

# show snmp

Use the **show snmp** command to display the system Simple Network Management Protocol (SNMP) current configuration. Variations of this command include the following:

**show snmp engineID**
**show snmp group**
**show snmp user**

## Syntax Description

| | |
|---|---|
| **engineID** | Displays the hex string that defines the current local engine ID settings. |
| **group** | Displays the list of all groups entered. |
| **user** | Displays the list of all users entered. |

## Default Values

No default is necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 13.1 | Command was expanded to include the **engineID**, **group**, and **user** options. |

## Usage Examples

The following is sample output using the **show snmp** command for a system with SNMP disabled and the default chassis and contact parameters:

```
>enable
#show snmp
Chassis: Chassis ID
Contact: Customer Service
0 Rx SNMP packets
  0 Bad community names
  0 Bad community uses
  0 Bad versions
  0 Silent drops
  0 Proxy drops
  0 ASN parse errors
```

The following is sample output of the **show snmp group** command for a situation in which a group called **securityV3auth** was defined (via the **snmp-server group** command) using version 3 and authentication, and no access control list:

>**enable**
**#show snmp group**

Group: securityV3auth Security Model: v3
Read View: default Write View: <not specified>
Notify View: default

# show sntp

Use the **show sntp** command to display the system Simple Network Time Protocol (SNTP) parameters and current status of SNTP communications.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Usage Examples

The following example displays the SNTP parameters and current status:

```
>enable
#show sntp
```

# show spanning-tree

Use the **show spanning-tree** command to display the status of the spanning-tree protocol. Variations of this command include:

**show spanning-tree**
**show spanning-tree** <*number*>
**show spanning-tree realtime**
**show spanning-tree** <*number*> **realtime**

> NOTE *Using the **realtime** argument for this command may adversely affect system performance and should be used with discretion.*

## Syntax Description

| | |
|---|---|
| <*number*> | Optional. Displays spanning-tree for a specific bridge group. This command is only applicable to routers configured for bridging. |
| **realtime** | Optional. Displays full-screen spanning tree information in real time. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

Use the **realtime** argument for this command to display full-screen output in real time. Information is continuously updated on the console until you either freeze the data (by pressing the **F** key) or exit **realtime** mode (by pressing **Ctrl-C**). If there is not enough room on the screen for all available data, the information will truncate at the bottom of the screen. In order to maximize the amount of data displayed, increase the terminal length (using the **terminal length** command; refer to *terminal length <number>* on page 410).

## Usage Examples

The following is an example output using the **show spanning-tree** command:

```
>enable
#show spanning-tree
Spanning Tree enabled protocol ieee
 Root ID    Priority    32768
            Address     00:a0:c8:00:88:41
            We are the root of the spanning tree
            Hello Time  2 sec Max Age 20 sec Forward Delay 15 sec
 Bridge ID Priority     32768
            Address     00:a0:c8:00:88:41
            Hello Time  2 sec Max Age 20 sec Forward Delay 15 sec
            Aging Time  300
```

| Interface | Role | Sts | Cost | Prio.Nbr | Type |
|---|---|---|---|---|---|
| eth 0/2 | Desg | FWD | 19 | 128.2 | P2p |
| eth 0/3 | Desg | FWD | 19 | 128.3 | P2p |
| eth 0/4 | Desg | FWD | 19 | 128.4 | P2p |
| giga-eth 0/1 | Desg | FWD | 4 | 128.25 | P2p |
| giga-eth 0/2 | Desg | FWD | 4 | 128.26 | P2p |

# show spanning-tree active

Use the **show spanning-tree active** command to display the spanning-tree status on active interfaces only. Variations of this command include:

**show spanning-tree active**
**show spanning-tree active detail**

## Syntax Description

| | |
|---|---|
| **detail** | Optional. Displays the spanning-tree protocol status in detail. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following is an example output using the **show spanning-tree active** command:

```
>enable
#show spanning-tree active
Spanning Tree enabled protocol ieee
  Root ID    Priority    32768
             Address     00:a0:c8:00:88:41
             We are the root of the spanning tree
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
Bridge ID Priority    32768
             Address     00:a0:c8:00:88:41
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time   300
eth 0/9         Desg FWD 19        128.9   P2p
eth 0/24        Desg FWD 19        128.24  P2p
Interface    Role    Sts    Cost    Prio.Nbr    Type
eth 0/2      Desg    FWD    19      128.2       P2p
eth 0/3      Desg    FWD    19      128.3       P2p
eth 0/9      Desg    FWD    19      128.9       P2p
```

# show spanning-tree blockedports

Use the **show spanning-tree blockedports** command to display ports that are currently in a blocked state.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following is an example output using the **show spanning-tree blockedports** command:

```
>enable
#show spanning-tree blockedports
Blocked Interfaces List
-----------------------------------
eth 0/3
giga-eth 0/2
p-chan 1
Number of blocked ports (segments) in the system: 3
```

# show spanning-tree interface <*interface*>

Use the **show spanning-tree interface** command to display spanning-tree protocol information for a particular interface. Variations of this command include:

**show spanning-tree interface** <*interface*>
**show spanning-tree interface** <*interface*> **active**
**show spanning-tree interface** <*interface*> **active detail**
**show spanning-tree interface** <*interface*> **cost**
**show spanning-tree interface** <*interface*> **edgeport**
**show spanning-tree interface** <*interface*> **priority**
**show spanning-tree interface** <*interface*> **rootcost**
**show spanning-tree interface** <*interface*> **state**

## Syntax Description

| | |
|---|---|
| <*interface*> | Specifies an interface in the format <*interface type [slot/port | slot/port.sub-interface id | interface id | interface id.sub-interface id]*>. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **show spanning-tree interface ?** for a complete list of valid interfaces. |
| **active** | Optional. Displays information for an active interface. |
| **active detail** | Optional. Displays detailed spanning-tree protocol information for an active interface. |
| **cost** | Optional. Displays only spanning-tree protocol path cost information. |
| **edgeport** | Optional. Displays information for all interfaces configured as edgeports. |
| **priority** | Optional. Displays only spanning-tree protocol priority information. |
| **rootcost** | Optional. Displays only spanning-tree protocol root path cost information. |
| **state** | Optional. Displays only spanning-tree protocol state information. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following is an example output using the **show spanning-tree interface ethernet** command:

```
>enable
#show spanning-tree interface ethernet 0/2
Interface         Role              Sts               Cost       Prio.Nbr    Type
----------------- ----------------- ----------------- ---------- ----------- ---------------
eth 0/2           Desg              LIS                19         128.2       P2p
```

# show spanning-tree pathcost method

Use the **show spanning-tree pathcost method** command to display the default pathcost method being used.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following is sample output using the **show spanning-tree pathcost method** command. In this case 32-bit values are being used when calculating path costs:

>**enable**
**#show spanning-tree pathcost method**
Spanning tree default pathcost method used is long

# show spanning-tree realtime

Use the **show spanning-tree realtime** command to display full-screen spanning tree information in real time. Variations of this command include:

**show spanning-tree realtime**
**show spanning-tree** <*number*> **realtime**

> NOTE *Using the **realtime** argument for this command can adversely affect the performance or your unit.*

## Syntax Description

| | |
|---|---|
| <*number*> | Optional. Displays spanning-tree for a specific bridge group. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |
| Release 10.1 | The real time display option was introduced. |

## Functional Notes

Use the **realtime** argument for this command to display full-screen output in real time. Information is continuously updated on the console until you either freeze the data (by pressing the **F** key) or exit **realtime** mode (by pressing **Ctrl-C**). If there is not enough room on the screen for all available data, the information will truncate at the bottom of the screen. In order to maximize the amount of data displayed, increase the terminal length (using the **terminal length** command; refer to *terminal length <number>* on page 410).

## Usage Examples

The following is sample output using the **show spanning-tree realtime** command.

```
>enable
#show spanning-tree realtime
STP 0
 Vlan 1
 Spanning Tree enabled protocol ieee 802.1w (Rapid Spanning-Tree)
 Root ID         Priority        8894
                 Address         00:a0:c8:00:f5:52
                 Cost            46
                 Port            1 (giga-eth 0/1)
                 Hello Time      2 sec Max Age 20 sec Forward Delay 15 sec
Bridge ID        Priority        32768
                 Address         00:a0:c8:02:f6:6b
                 Aging Time      300
Interface        Role        Sts         Cost       Prio.Nbr       Type
---------------- ---- --- --------- -------- --------------------------------------------------------
giga-eth 0/1     Root        FWD         19         128.1          P2p
giga-eth 0/5     Altn        BLK         19         128.5          P2p
-------------------------------------------------------------------------------------------------------
 Exit - 'Ctrl-C', Freeze - 'f', Resume - 'r'
```

# show spanning-tree root

Use the **show spanning-tree root** command to display information regarding the spanning-tree protocol root. Variations of this command include:

**show spanning-tree root**
**show spanning-tree root address**
**show spanning-tree root cost**
**show spanning-tree root detail**
**show spanning-tree root forward-time**
**show spanning-tree root hello-time**
**show spanning-tree root id**
**show spanning-tree root max-age**
**show spanning-tree root port**
**show spanning-tree root priority**
**show spanning-tree root priority system-id**

## Syntax Description

| | |
|---|---|
| **address** | Optional. Displays the address of the spanning-tree root. |
| **cost** | Optional. Displays the path cost of the spanning-tree root. |
| **detail** | Optional. Displays the spanning-tree root information in detail. |
| **forward-time** | Optional. Displays the forward-time of the spanning-tree root. |
| **hello-time** | Optional. Displays the hello-time of the spanning-tree root. |
| **id** | Optional. Displays the ID of the spanning-tree root. |
| **max-age** | Optional. Displays the maximum age of the spanning-tree root. |
| **port** | Optional. Displays the port of the spanning-tree root. |
| **priority** | Optional. Displays the priority of the spanning-tree root. |
| **priority system-id** | Optional. Displays the priority and system-id of the spanning-tree root. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following is an example output using the **show spanning-tree root** command:

```
>enable
#show spanning-tree root
Root ID                 Root Cost   Hello Time  Max Age    Fwd Dly     Root Port
----------------------  ----------  ----------  ---------- ----------  ----------
8191 00:a0:c8:b9:bb:82  108         2           20         15          eth 0/1
```

# show stack

Use the **show stack** command to view the status of all the switches configured for stacking. Displays the mode of the switch as either master or member. If the mode is master, this command also gives the status of the stack members. Variations of this command include:

**show stack**
**show stack candidates**
**show stack candidates realtime**
**show stack realtime**
**show stack topology**
**show stack topology realtime**

*Using the **realtime** argument for this command can adversely affect the performance or your unit.*

## Syntax Description

| | |
|---|---|
| **candidates** | Optional. Displays all units that have registered with this stack master. This option is only available on a switch configured as a stack master. |
| **topology** | Optional. Displays the stack topology. This option is only available on a switch configured as a stack master. |
| **realtime** | Optional. Displays full-screen output in real time. Refer to *Functional Notes* below for more information. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |
| Release 10.1 | The real time display option was introduced. |

## Functional Notes

The stack candidates are a list of units that could be added to the stack. They are not yet members.

Use the **realtime** argument for this command to display full-screen output in real time. Information is continuously updated on the console until you either freeze the data (by pressing the **F** key) or exit **realtime** mode (by pressing **Ctrl-C**). If there is not enough room on the screen for all available data, the information will truncate at the bottom of the screen. In order to maximize the amount of data displayed, increase the terminal length (using the **terminal length** command; refer to *terminal length <number>* on page 410).

## Usage Examples

The following example displays the configuration of the switch stack while in stack-master mode:

```
>enable
#show stack
Stack mode is MASTER
 Management Vlan is 2386, firmware version is 08.00.18.D
 Stack network is 169.254.0.0/24

Stack members...

Member  Mac Address         Mgmt IP Address  Source Interface  State
2       00:A0:C8:02:CF:C0   169.254.0.2      Stack port        Up
3       00:A0:C8:00:8C:20   169.254.0.3      Stack port        Up
#
```

| | |
|---|---|
| **Member** | Specifies the stack member's Unit ID. |
| **MAC Address** | Specifies the stack member's MAC address. |
| **Mgmt IP Address** | Specifies the stack member's IP address. |
| **Source Interface** | Specifies the interface that the stack member was learned from. |
| **State** | Specifies the stack member's state: **Up** (member is up and functioning properly); **Down** (member was at one time functioning, but we have lost contact with it); **Waiting** (we are waiting for the unit to register with us; when he does, we will add it to the stack); **Denied** (the unit could not be added to the stack because the stack protocol versions were not compatible). |

The following example displays the configuration of the switch stack while in stack-member mode:

```
->enable
#show stack

Stack mode is STACK-MEMBER
 My Unit ID is 3, management Vlan is 2386
 Stack management network is 169.254.0.0/24

Stack Master info:
 Master is “Switch”, learned via giga-eth 0/1
 IP address is 169.254.0.1, MAC address is 00:DE:AD:00:65:83
#
```

The following example displays all units that have registered with this stack-master:

->**enable**
**#show stack candidates**
-Displaying all known Stack candidates...

| MAC Address | System Name | Source Interface | AOS Revision |
|---|---|---|---|
| 00:A0:C8:00:8C:20 | LabSwitch1 | stack port | 08.00.18 |
| 00:A0:C8:00:F5:6C | LabSwitch2 | stack port | 08.00.19.D |
| 00:A0:C8:02:CF:C0 | LabSwitch3 | stack port | 08.00.20.D |

#

# show startup-config

Use the **show startup-config** command to display a text printout of the startup configuration file stored in NVRAM. Variations of this command include:

**show startup-config**
**show startup-config checksum**

## Syntax Description

| | |
|---|---|
| **checksum** | Displays the Message Digest 5 (MD5) checksum of the unit's startup configuration. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

This command is used in conjunction with the **show running-config checksum** command to determine whether the configuration has changed since the last time it was saved.

## Usage Examples

The following is a sample output of the **show startup-config** command:

```
>enable
#show startup-config
!
!
no enable password
!
ip subnet-zero
ip classless
ip routing
!
event-history on
no logging forwarding
logging forwarding priority-level info
no logging email
!
ip policy-timeout tcp all-ports 600
ip policy-timeout udp all-ports 60
ip policy-timeout icmp 60
!
```

```
!
!
interface eth 0/1
speed auto
  no ip address
  shutdown
!
interface dds 1/1
  shutdown
!
interface bri 1/2
  shutdown
!
!
ip access-list standard MatchAll
  permit host 10.3.50.6
  permit 10.200.5.0 0.0.0.255
!
!
ip access-list extended UnTrusted
  deny   icmp 10.5.60.0 0.0.0.255 any source-quench
  deny   tcp any any
!
no ip snmp agent
!
!
!
```

# show system

The **show system** command shows the system version, timing source, power source, and alarm relay status.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Usage Examples

The following is sample output for the **show system** command:

```
>enable
#show system
ADTRAN, Inc. OS version 07.00.20
 Checksum: 3B2FCC0F, built on Tue Jun 01 13:36:36 2004
Boot ROM version 07.00.20
 Checksum: 604D, built on: Tue Jun 01 13:59:11 2004
Copyright (c) 1999-2004, ADTRAN, Inc.
Platform: Total Access 900
Serial number TechPub
Flash: 8388608 bytes DRAM: 33554431 bytes
ICP uptime is 0 days, 0 hours, 53 minutes, 50 seconds
System returned to ROM by External Hard Reset
Current system image file is “070020.biz”
Boot system image file is “070020.biz”
Power Source: AC
Primary System clock source config: t1 0/1
Secondary System clock source config: t1 0/1
Active System clock source: t1 0/1
Alarm Relay: OPEN
```

# show tacacs+ statistics

Use the **show tacacs+ statistics** command to display terminal access controller access control system (TACACS+) client statistics.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following is sample output for the **show tacacs+ statistics** command:

```
>enable
#show tacacs+ statistics
```

| | Authentication | Authorization | Accounting |
|---|---|---|---|
| Packets sent: | 0 | 0 | 0 |
| Invalid responses: | 0 | 0 | 0 |
| Timeouts: | 0 | 0 | 0 |
| Average delay: | 0ms | 0ms | 0ms |
| Maximum delay: | 0ms | 0ms | 0ms |
| Socket Opens: | 0 | | |
| Socket Closes: | 0 | | |
| Socket Aborts: | 0 | | |
| Socket Errors: | 0 | | |
| Socket Timeouts: | 0 | | |
| Socket Failed Connections: | 0 | | |
| Socket Packets Sent: | 0 | | |
| Socket Packets Received: | 0 | | |

# show tcp info

Use the **show tcp info** command to display Transmission Control Protocol (TCP) control block information in AOS. This information is for troubleshooting and debug purposes only. For more detailed information, you can optionally specify a particular TCP control block. When a particular TCP control block is specified, the system provides additional information regarding crypto map settings that the **show tcp info** command does not display. Variations of this command include:

**show tcp info**
**show tcp info realtime**
**show tcp info** <*control block*>
**show tcp info** <*control block*> **realtime**


*Using the **realtime** argument for this command can adversely affect the performance or your unit.*

## Syntax Description

| | |
|---|---|
| <*control block*> | Optional. Specifies a particular TCP control block for more detailed information. |
| **realtime** | Optional. Displays full-screen output in real time. Refer to the *Functional Notes* below for more information. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 4.1 | Command was introduced. |
| Release 10.1 | The real time display option was introduced. |

## Function Notes

Use the **realtime** argument for this command to display full-screen output in real time. Information is continuously updated on the console until you either freeze the data (by pressing the **F** key) or exit **realtime** mode (by pressing **Ctrl-C**). If there is not enough room on the screen for all available data, the information will truncate at the bottom of the screen. In order to maximize the amount of data displayed, increase the terminal length (using the **terminal length** command; refer to *terminal length <number>* on page 410).

## Usage Examples

The following is a sample from the **show tcp info** command:

>**enable**
**#show tcp info**
TCP TCB Entries

| ID | STATE | LSTATE | OSTATE | TYPE | FLAGS | RPORT | LPORT | SWIN | SRT | INTERFACE |
|---|---|---|---|---|---|---|---|---|---|---|
| 0 | FREE | FREE | FREE | SRVR | 0 | 0 | 0 | 0 | 0 | NONE |
| 1 | LISTEN | | FREE | FREE | CONN | 0 | 0 | 21 | 0 | 0NONE |
| 2 | LISTEN | FREE | FREE | CONN | 0 | 0 | 80 | 0 | 0 | NONE |
| 3 | LISTEN | FREE | FREE | CONN | 0 | 0 | 23 | 0 | 0 | NONE |
| 4 | LISTEN | FREE | FREE | CONN | 0 | 0 | 5761 | 0 | 0 | NONE |
| 5 | FREE | FREE | FREE | SRVR | 0 | 0 | 0 | 0 | 0 | NONE |

etc.

# show tech

Use the **show tech** command to save technical information to a file named showtech.txt. Variations of this command include:

**show tech**
**show tech terminal**

## Syntax Description

| | |
|---|---|
| **terminal** | Optional. Displays the **showtech.txt** file output tot he terminal screen in real time. |

## Default Values

No default necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Functional Notes

The **show tech** command runs a script that creates a **showtech.txt** file in flash memory that contains the command output from the following show commands:

*Not all listed **show** commands apply to all ADTRAN products.*

**show version**
**show modules**
**show flash**
**show cflash**
**show running-config verbose**
**show interfaces**
**show atm pvc**
**show dial-backup interfaces**
**show frame-relay lmi**
**show frame-relay pvc**
**show ip bgp neighbors**
**show ip bgp summary**
**show ip ospf neighbor**
**show ip ospf summary-address**
**show ip mroute**
**show ip bridge**
**show spanning-tree**
**show ip interfaces**

**show connections**
**show arp**
**show ip traffic**
**show tcp info**
**show ip protocols**
**show ip route**
**show ip access-lists**
**show event-history**
**show output-startup**
**show processes cpu**
**show buffers**
**show buffers users**
**show memory heap**
**show debugging**

## Usage Examples

The following example creates a **showtech.txt** file and displays it to the terminal screen:

>**enable**
**#show tech**
Opening and applying file.....
Done.

# show temperature

Use the **show temperature** command to display the unit temperature.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |

## Usage Examples

The following is sample output from the **show temperature** command:

```
>enable
#show temperature

Temperature: 33 degrees C
```

# show thresholds

Use the **show thresholds** command to display thresholds currently crossed for all DS1 interfaces.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following is sample output of the **show thresholds** command.

```
>enable
#show thresholds
t1 1/1:
    SEFS 15 min threshold exceeded
    UAS 15 min threshold exceeded
    SEFS 24 hr threshold exceeded
    UAS 24 hr threshold exceeded
t1 1/2:
    No thresholds exceeded
```

# show toneservices resources

Use the **show toneservices resources** command to display Digital Signal Processor (DSP) tone information.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following is sample output for this command:

```
>enable
#show toneservices resources

DSP   Channel  Type    Port     Status
-------------------------------------------------------------
 0/1    1        RTP    N/A      Available
 0/1    2        RTP     N/A     Available
 0/1    3        RTP    N/A      Available
 0/1    4        RTP    N/A      Available
 0/1    5        RTP    N/A      Available
 0/1    6        RTP    N/A      Available
 0/1    7        RTP    N/A      Available
 0/1    8        RTP    N/A      Available
 0/1    9        RTP    N/A      Available
 0/1    10       RTP    N/A      Available
 0/1    11       RTP    N/A      Available
 0/1    12       RTP    N/A      Available
 0/1    13       RTP    N/A      Available
 0/1    14       RTP    N/A      Available
 0/1    15       RTP    N/A      Available
```

# show track

Use the **show track** command to display track object configuration and statistics. Refer to *Network Monitor Track Configuration Command Set* on page 1698 for information on configuring track objects. Variations of this command include the following:

**show track**
**show track** <*name*>
**show track** <*name*> **realtime**

## Syntax Description

| | |
|---|---|
| <*name*> | Optional. Displays information only for the track object specified. |
| **realtime** | Optional. Displays full-screen output in real time. Refer to the *Functional Notes* below for more information. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Functional Notes

Use the **realtime** argument for this command to display full-screen output in real time. Information is continuously updated on the console until you either freeze the data (by pressing the **F** key) or exit **realtime** mode (by pressing **Ctrl-C**). If there is not enough room on the screen for all available data, the information will truncate at the bottom of the screen. In order to maximize the amount of data displayed, increase the terminal length (using the **terminal length** command; refer to *terminal length <number>* on page 410).

## Usage Examples

The following is a sample output of the **show track** command:

```
>enable
#show track track_1
Current State: PASS
    Dampening Interval: 30 seconds
    Test Value: probe_A (PASS) AND probe_B (FAIL)
    Track Changes: 3
    Time in current state: 25 days 2 hours, 34 minutes, 32 seconds
```

# show udp info

Use the **show udp info** command to display User Datagram Protocol (UDP) session information. Variations of this command include:

**show udp info**
**show udp info realtime**
**show udp info** <*number*>
**show udp info** <*number*> **realtime**


*Using the **realtime** argument for this command can adversely affect the performance or your unit.*

## Syntax Description

| | |
|---|---|
| <*number*> | Optional. Specifies ID of session to display. Valid range is 0 to 31. |
| **realtime** | Optional. Displays full-screen output in real time. Refer to the *Functional Notes* below for more information. |

## Default Values

No default necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |
| Release 10.1 | The real time display option was introduced. |

## Functional Notes

Use the **realtime** argument for this command to display full-screen output in real time. Information is continuously updated on the console until you either freeze the data (by pressing the **F** key) or exit **realtime** mode (by pressing **Ctrl-C**). If there is not enough room on the screen for all available data, the information will truncate at the bottom of the screen. In order to maximize the amount of data displayed, increase the terminal length (using the **terminal length** command; refer to *terminal length <number>* on page 410).

## Usage Examples

The following example shows sample output from the **show udp info** command:

```
>enable
#show udp info
UDP Session Entries
ID      Local Port      IP Address      Socket
---     ---------------  -----------------  -----------
2       520             0.0.0.0         1
3       0               0.0.0.0         4
4       161             0.0.0.0         5
5       8               127.0.0.1       7
6       10              0.0.0.0         11
7       6               127.0.0.1       16
8       4               127.0.0.1       17
9       14              127.0.0.1       18
10      12              127.0.0.1       19
```

# show users

Use the **show users** command to display the name (if any) and state of users authenticated by the system. Variations of this command include:

**show users**
**show users realtime**


*Using the **realtime** argument for this command can adversely affect the performance or your unit.*

## Syntax Description

| | |
|---|---|
| **realtime** | Optional. Displays full-screen output in real time. Refer to the *Functional Notes* below for more information. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |
| Release 10.1 | The real time display option was introduced. |

## Functional Notes

Use the **realtime** argument for this command to display full-screen output in real time. Information is continuously updated on the console until you either freeze the data (by pressing the **F** key) or exit **realtime** mode (by pressing **Ctrl-C**). If there is not enough room on the screen for all available data, the information will truncate at the bottom of the screen. In order to maximize the amount of data displayed, increase the terminal length (using the **terminal length** command; refer to *terminal length <number>* on page 410).

Displayed information includes:

- Connection location (for remote connections this includes Transmission Control Protocol (TCP) information)
- user name of authenticated user
- Current state of the login (in process or logged in)
- Current enabled state
- Time the user has been idle on the connection

## Usage Examples

The following is a sample of **show users** output:

```
>enable
#show users
- CONSOLE 0 'adtran' logged in and enabled
  Idle for 00:00:00
- TELNET 0 (172.22.12.60:3998) 'password-only' logged in (not enabled)
  Idle for 00:00:14
- FTP (172.22.12.60:3999) 'adtran' logged in (not enabled)
  Idle for 00:00:03
```

# show version

Use the **show version** command to display the current ADTRAN operating system (AOS) version information.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following is a sample **show version** output:

```
>enable
#show version

AOS version: 02.01.00
Checksum: 1505165C Built on: Fri Aug 23 10:23:13 2002
 Upgrade key: 420987gacs9097gbdsado
BootROM version: 02.01.00
 Checksum: DB85 Built on: Mon Aug 19 10:33:03 2002
Copyright 1999-2002 ADTRAN Inc.
Serial number b104

Router uptime is 0 days 3 hours 9 minutes 54 seconds
System returned to ROM by External Hard Reset
System image file is “020100.biz”
```

# show vlan

Use the **show vlan** command to display current virtual local area network (VLAN) information. Variations of this command include:

**show vlan**
**show vlan brief**
**show vlan brief realtime**
**show vlan id** <*vlan id*>
**show vlan name** <*name*>
**show vlan realtime**

*Using the **realtime** argument for this command can adversely affect the performance or your unit.*

## Syntax Description

| | |
|---|---|
| **brief** | Optional. Shows an abbreviated version of the VLAN information (brief description). |
| **id** <*vlan id*> | Optional. Shows information regarding a specific VLAN, specified by a VLAN interface ID (valid range: 1 to 4094). |
| **name** <*name*> | Optional. Shows information regarding a specific VLAN, specified by a VLAN interface name (up to 32 characters). |
| **realtime** | Optional. Displays full-screen output in real time. Refer to the *Functional Notes* below for more information. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |
| Release 10.1 | The real time display option was introduced. |

## Function Notes

Use the **realtime** argument for this command to display full-screen output in real time. Information is continuously updated on the console until you either freeze the data (by pressing the **F** key) or exit **realtime** mode (by pressing **Ctrl-C**). If there is not enough room on the screen for all available data, the information will truncate at the bottom of the screen. In order to maximize the amount of data displayed, increase the terminal length (using the **terminal length** command; refer to *terminal length <number>* on page 410).

## Usage Examples

The following is a sample **show vlan** output:

```
>enable
#show vlan
```

| VLAN | Name | Status | Ports |
|---|---|---|---|
| - 1 | Default | active | eth 0/5, eth 0/6, eth 0/8, eth 0/13, eth 0/14, eth 0/15, eth 0/16, eth 0/17, eth 0/18, eth 0/19, eth 0/20, eth 0/21, eth 0/22, eth 0/23, eth 0/24, giga-eth 0/1, giga-eth 0/2 |
| 2 | accounting | active | eth 0/1, eth 0/2 |
| 3 | VLAN0003 | active | eth 0/3, eth 0/4, eth 0/7, eth 0/9, eth 0/10, eth 0/11, eth 0/12 |

| VLAN | Type | MTU |
|---|---|---|
| - 1 | enet | 1500 |
| 2 | enet | 1500 |
| 3 | | |

The following is an example of the **show vlan name** command that displays VLAN 2 (**accounting** VLAN) information:

```
>enable
#show vlan name accounting
```

| VLAN | Name | Status | Ports |
|---|---|---|---|
| - 2 | accounting | active | eth 0/1, eth 0/2 |

| VLAN | Type | MTU |
|---|---|---|
| - 2 | enet | 1500 |

# show voice alias

Use the **show voice alias** command to display alias parameters. Aliases are used to mask identity settings such as names and extensions. Variations of this command include:

**show voice alias**
**show voice alias global**
**show voice alias group**
**show voice alias system**
**show voice alias user**

## Syntax Description

| | |
|---|---|
| **global** | Optional. Displays global aliases. |
| **group** | Optional. Displays group aliases. |
| **system** | Optional. Displays system aliases. |
| **user** | Optional. Displays user aliases. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example shows sample output from the **show voice alias** command:

>**enable**
**#show voice alias**

```
Alias           Translation     Type
--------------- --------------- ------------------
MyAlias         4433            Global

Total Displayed: 1
```

# show voice ani

Use the **show voice ani** command to display voice automatic number identification (ANI) substitution parameters. Variations of this command include:

**show voice ani**
**show voice ani match** <*substitution*>

## Syntax Description

| | |
|---|---|
| **match** <*substitution*> | Optional. Displays a specific ANI substitution entry. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Functional Notes

Examples and rules of use.

| MATCH # | SUBST # |
|---|---|
| 1. 256-963-XXXX | 9-963-XXXX |
| 2. NXX-NXX-XXXX | 9-1-NXX-NXX-XXXX |
| 3. $ | 9$ |

MATCH Number Rules -
1. All "," characters are ignored.
2. All "[" and "]" brackets must match and contain numbers only [123].
3. If using a "$" wildcard, it is the only character allowed.
4. "X" matches [0-9], "N" matches [2-9].

## Usage Examples

The following example displays voice ANI information:

```
>enable
#show voice ani
```

# show voice available

Use the **show voice available** command to list foreign exchange station (FXS) ports that are not associated with a user.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example displays FXS ports that are not associated with a user:

```
>enable
#show voice available
```

# show voice dial-plan

Use the **show voice dial-plan** command to view number display templates. Variations of this command include:

**show voice dial-plan**
**show voice dial-plan** *<number>*

## Syntax Description

| | |
|---|---|
| *<number>* | Optional. Displays information about a specific number display template. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example displays information about number display template **1**:

```
>enable
#show voice dial-plan 1
Type                    ID      Pattern
-------------------------------------------------------------------
Always Permitted        1       NXXNXXXXXX
```

# show voice did

Use the **show voice did** command to display Direct Inward Dialing (DID) information. Variations of this command include:

**show voice did**
**show voice did groups**
**show voice did other**
**show voice did users**

## Syntax Description

| | |
|---|---|
| **groups** | Optional. Displays all DID entries for ring groups. |
| **other** | Optional. Displays all non-user and non-ring group DID entries. |
| **users** | Optional. Displays all DID entries for users. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example displays DID entries for ring groups:

>**enable**
**#show voice did groups**

# show voice directory

Use the **show voice directory** command to display Direct Inward Dialing (DID) information. Variations of this command include:

**show voice directory**
**show voice directory extension**
**show voice directory last**

## Syntax Description

| | |
|---|---|
| **extension** | Optional. Displays all extensions sorted by extension number. |
| **last** | Optional. Displays all extensions sorted by the user's last name. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example displays all extensions sorted by extension number:

```
>enable
#show voice directory extension
Ext     Name
-----------------------------------
5200    Smith, Bill
6000    Peters, Scott
6001    Jones, Eve
6002    Patterson, Bill
6003    Scott, Julie
```

# show voice door-phone

Use the **show voice door-phone** command to display the door phone account settings. A door phone is used to communicate with visitors prior to them entering an establishment.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example shows sample output from the **show voice door-phone** command:

```
>enable
#show voice door-phone

First       Last          Ext   Interface        Description
-------------------------------------------------------------------------------
Front       Door          4430  virtual          Front Door of Building
```

# show voice extensions

Use the **show voice extensions** command to display all of the current voice extensions and their status.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example displays all extensions and the status of the extension:

```
>enable
#show voice directory extension

AccountID       Idle/Ring/Busy    Available   DND     FWD
-------------------------------------------------------------------------------
T01             Idle              *           -       -
T06             Idle              *           -       -
T02             Idle              *           -       -
5200            Idle              *           -       -
6000            Idle              *           -       -
6001            Idle              *           -       -
6002            Idle              *           -       -
6003            Idle              *           -       -
T03             Idle              *           -       -
2               Idle              *           -       -
1234            Idle              *           -       -
```

# show voice grouped-trunk

Use the **show voice grouped-trunk** command to display all voice trunk groups.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example displays all voice trunk groups:

```
>enable
#show voice trunked-group

Name        Resource-Selection      Description
-------------------------------------------------------------------------
SIP         linear                  SIP trunk
DSX         linear                  DSX trunk
DXS         linear                  DXS trunk
```

# show voice mail

Use the **show voice mail** command to display voice mail information. Variations of this command include:

**show voice mail**
**show voice mail** <*number*>
**show voice mail notify-schedule** <*number*>

## Syntax Description

| | |
|---|---|
| <*number*> | Optional. Displays voice mail information for the specified user's extension. |
| **notify-schedule** <*number*> | Optional. Specifies the extension the voice mail notification to display. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example shows sample output from the **show voice mail** command:

**>enable**
**#show voice mail**

| AccountID | VM COS | New Msg | Num Msg | Total Time Used (min) | Total Time Free (min) | Greeting Time (min) |
|---|---|---|---|---|---|---|
| 5000 | VM-ALLOWED | - | - | 00:00 | 10:00 | 00:00 |
| 5001 | VM-ALLOWED | - | - | 00:00 | 10:00 | 00:00 |

# show voice phone-files

Use the **show voice phone-files** command to display files required for Session Initiation Protocol (SIP) phone configuration.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example shows sample output from the **show voice phone-files** command:

```
>enable
#show voice phone-files

03/02/2006 16:03 PM             216 0004f203f69c.cfg
03/02/2006 16:03 PM             687 5001-0004f203f69c.cfg
03/02/2006 16:03 PM             216 0004f203b0d6.cfg
03/14/2006 16:03 PM             306 polycom.cfg
7 File(s)          132516 bytes
0 Dir(s)           0 bytes
21019575 bytes used, 9720360 available, 30739935 total
```

# show voice quality-stats

Use the **show voice quality-stats** command to display voice quality statistical information. Variations of this command include the following:

**show voice quality-stats**
**show voice quality-stats active**
**show voice quality-stats active realtime**
**show voice quality-stats** <*ID*>
**show voice quality-stats** <*ID*> **realtime**

*Using the **realtime** argument for this command can adversely affect the performance or your unit.*

## Syntax Description

| | |
|---|---|
| **active** | Displays all quality statistics for active calls. |
| <*ID*> | Specifies an identity number of a call to obtain detail statistics of a specific call. |
| **realtime** | Optional. Displays full-screen output in realtime. Refer to the *Functional Notes* below for more information. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Functional Notes

Use the **realtime** argument for this command to display full-screen output in real time. Information is continuously updated on the console until you either freeze the data (by pressing the **F** key) or exit **realtime** mode (by pressing **Ctrl-C**). If there is not enough room on the screen for all available data, the information will truncate at the bottom of the screen. In order to maximize the amount of data displayed, increase the terminal length (using the **terminal length** command; refer to *terminal length <number>* on page 410).

## Usage Examples

The following example displays voice quality statistics for all active calls:

>**enable**
**#show voice quality-stats active**

# show voice ring-group

Use the **show voice ring-group** command to display all ring groups. Variations of this command include:

**show voice ring-group**
**show voice ring-group** <*number*>

## Syntax Description

| | |
|---|---|
| <*number*> | Optional. Displays information about a specific ring group extension. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example displays all ring groups:

>**enable**
**#show voice ring-group**
ring-group 1234 type: linear
description:
Number of calls allowed: 1

| First | Last | Ext | Logged In |
|---|---|---|---|

| Order | NumRings | Action |
|---|---|---|
| 1 | 2 | None |
| 2 | 2 | None |
| 3 | 2 | None |
| 4 | 2 | None |
| 5 | 2 | None |

ring-group 2 type: linear
description:
Number of calls allowed: 1

| First | Last | Ext | Logged In |
|---|---|---|---|

| Order | NumRings | Action |
|---|---|---|
| 1 | 2 | None |
| 2 | 2 | None |
| 3 | 2 | None |

# show voice service-mode

Use the **show voice service-mode** command to display all voice service mode transitions.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example displays all voice service mode transitions:

```
>enable
#show voice service-mode
```

# show voice speed-dial

Use the **show voice speed-dial** command to display system speed dial information. Variations of this command include:

**show voice speed-dial**
**show voice speed-dial** <*number*>

## Syntax Description

| | |
|---|---|
| <*number*> | Optional. Displays information about a specific speed dial number. Valid range is 1 to 99. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example displays information on speed dial number **50**:

>**enable**
**#show voice speed-dial 50**
speed-dial - ID: 50
Name: Main Office
Number: 4000

# show voice spre

Use the **show voice spre** command to display all special prefix (SPRE) codes. Variations of this command include:

**show voice spre local**
**show voice spre network**

## Syntax Description

| | |
|---|---|
| **local** | Displays all SPRE codes used locally. |
| **network** | Displays all SPRE codes passed through to the network. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example displays all local SPRE codes:

**>enable**
**#show voice spre local**

# show voice switchboard

Use the **show voice switchboard** command to display all voice switchboard extensions.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example displays all voice switchboard extensions:

```
>enable
#show voice switchboard
Ext
----
1234
2
5200
6000
6001
6002
6003
```

# show voice trunk

Use the **show voice trunk** command to display all voice trunks. Variations of this command include:

**show voice trunk**
**show voice trunk** *<trunk id>*
**show voice trunk connects** *<trunk id>*

## Syntax Description

| | |
|---|---|
| **connects** | Optional. Displays all trunk voice interface connections. |
| *<trunk id>* | Optional. Displays voice trunk information for a specific trunk ID. Use T01, T02, and so on for the trunk ID. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example displays all voice trunks:

**>enable**
**#show voice trunk**

| Trunk Name | Resource Selection | Busy Admin. Config. | Busy Admin. Status | Busy Attempts Today | Non Busy Attempts Today | Busy Attempts Total | Non Busy Attempts Total |
|---|---|---|---|---|---|---|---|
| T01 | linear | Not Busy | Not Busy | 0 | 0 | 0 | 7 |
| T06 | linear | Not Busy | No Connects | 0 | 0 | 0 | 0 |
| T02 | linear | Not Busy | Not Busy | 0 | 0 | 0 | 27 |
| T03 | linear | Not Busy | No Connects | 0 | 0 | 0 | 0 |

The following example displays all voice interface connections:

**>enable**
**#show voice switchboard**

| Trunk | Interface | Slot/Port | Tdm-group | Timeslot |
|---|---|---|---|---|
| T02 | t1 | 0/2 | 1 | 1-24 |

# show voice users

Use the **show voice users** command to display all voice user stations. Variations of this command include:

**show voice users**
**show voice users did**

## Syntax Description

| | |
|---|---|
| **did** | Optional. Displays all users with Direct Inward Dial (DID) extensions. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example displays all voice users:

>**enable**
**#show voice users**

| First | Last | Ext | Interface Description |
|---|---|---|---|
| Janet | Smith | 5200 | virtual |
| Bill | Jones | 6000 | virtual |
| Sam | Sampson | 6001 | virtual |
| Eve | Smith | 6002 | virtual |
| Bob | Wilson | 6003 | virtual |

# sip check-sync

Use the **sip check-sync** command to send a check-sync notification to all IP phones registered to the unit. When an IP phone receives this check-sync notification, the phone will check for possible configuration changes stored on the server. Variations of this command include the following:

**sip check-sync**
**sip check-sync firmware upgrade**
**sip check-sync** *<user name or ip address>*

## Syntax Description

| | |
|---|---|
| **firmware-upgrade** | Specifies an check-sync to be used when upgrading phone firmware. |
| *<user name or ip address>* | Specifies a specify phone to notify of configuration changes. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example notifies all IP phones to check for a change in configuration:

**>enable**
**#sip check-sync**

# **telnet** *<ip address>*

Use the **telnet** command to open a Telnet session (through AOS) to another system on the network.

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies the IP address of the remote system. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example opens a Telnet session with a remote system (**10.200.4.15**):

>**enable**
**#telnet 10.200.4.15**
User Access Login:
Password:

# telnet stack-member <*unit id*>

Use the **telnet stack-member** command to Telnet to a stack member.

## Syntax Description

| | |
|---|---|
| <*unit id*> | Specifies unit ID of the stack member to connect to via a Telnet session. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Functional Notes

This command is only available when in stack-master mode.

## Usage Examples

The following example Telnets to a member of the stack:

>**enable**
**#telnet stack-member 3**
Trying Stack Member 3...Press Ctrl+C to abort

# terminal length <*number*>

The **terminal length** command sets the number of rows (lines) for a terminal session. Use the **no** form of this command to return to the default value. This command is only valid for the current session and returns to the default (24 rows) when the session closes.

## Syntax Description

| | |
|---|---|
| <*number*> | Specifies the number of rows for a terminal session. Range is 0 to 480 lines. Setting the **terminal length** to 0 disables paging. |

## Default Values

The default setting for this command is 24 rows.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Usage Examples

The following example sets the number of rows for a terminal session to **30**.

```
>enable
#terminal length 30
```

# traceroute

Use the **traceroute** command to display the IP routes a packet takes to reach the specified destination. Variations of this command include:

**traceroute**
**traceroute** <*ip address*>
**traceroute** <*ip address*> **source** <*ip address*>

## Syntax Description

| | |
|---|---|
| <*ip address*> | Optional. Specifies the IP address of the remote system's route to trace. |
| **source** <*ip address*> | Optional. Specifies the IP address of the interface to use as the source of the trace.<br>IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following is a sample **traceroute** output:

```
>enable
#traceroute 192.168.0.1

Type CTRL+C to abort.
Tracing route to 192.168.0.1 over a maximum of 30 hops
  1    22ms   20ms   20ms    192.168.0.65
  2    23ms   20ms   20ms    192.168.0.1
#
```

# undebug all

Use the **undebug all** command to disable all activated debug messages.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |

## Usage Examples

The following example disables all activated debug messages:

```
>enable
#undebug all
```

# vlan database

Use the **vlan database** command to enter the VLAN Database Configuration mode. Refer to the section *VLAN Database Configuration Command Set* on page 1533 for more information.

## Syntax Description

No subcommands.

## Default Values

No defaults necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following example enters the VLAN Configuration mode:

```
>enable
#vlan database
```

# voice email

Use the **voice email** command to send email messages to the system's voice users. To cancel the email, enter '**..**'. Variations of this command include:

**voice email all**
**voice email** *<space-separated list>*

## Syntax Description

| | |
|---|---|
| **all** | Sends an email to all Voice users. |
| *<space-separated list>* | Sends an email to specific Voice users (addresses and/or extensions). |

## Default Values

No default values necessary for this command.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example sends an email to **all** voice users:

```
#voice email all
    Enter your email address (or extension).
    Email Sender: Admin
    Enter space-separated list of filenames to attach to email.
      Enter '–' for none.
    Email Attachments: –
    Email Subject: Change Passwords
      Enter '–' for a blank line.
      Enter a '.' on a line by itself to end.
      Enter a '..' to cancel email.
    Email Body: Your password will expire in 2 days. Please change your password.
    Email Body:.
    Email from Admin
    Sending to Spa (Spa).
```

# **wall** *<message>*

Use the **wall** command to send messages to all users currently logged in to AOS unit.

## Syntax Description

| | |
|---|---|
| *<message>* | Sends a message to all user's logged in to the command line interface (CLI). |

## Default Values

No defaults necessary for this command.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Usage Examples

The following example sends the message "Reboot in 5 minutes if no objections" to the CLI screen of everyone currently connected:

**>enable**
**#wall Reboot in 5 minutes if no objections**

# write

Use the **write** command to save the running configuration to the unit's nonvolatile random access memory (NVRAM) or a Trivial File Transfer Protocol (TFTP) server. Also use the **write** command to clear NVRAM or to display the running configuration on the terminal screen. Entering the **write** command with no other arguments copies your configuration changes to the unit's NVRAM. Once the save is complete, the changes are retained even if the unit is shut down or suffers a power outage. Variations of this command include:

**write**
**write dynvoice-config**
**write erase**
**write memory**
**write network**
**write terminal**

## Syntax Description

| | |
|---|---|
| **dynvoice-config** | Optional. Writes dynvoice configuration information to the unit's NVRAM. |
| **erase** | Optional. Erases the configuration files saved to the unit's NVRAM. |
| **memory** | Optional. Saves the current configuration to NVRAM. Refer to *copy <source> <destination>* on page 95 for more information. |
| **network** | Optional. Saves the current configuration to the network TFTP server. Refer to *copy tftp <destination>* on page 99 for more information. |
| **terminal** | Optional. Displays the current configuration on the terminal screen. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Usage Examples

The following example saves the current configuration to the unit's NVRAM:

```
>enable
#write memory
```

# GLOBAL CONFIGURATION MODE COMMAND SET

To activate the Global Configuration mode, enter the **configuration** command at the Enable mode prompt. For example:

```
>enable
#configure terminal
(config)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:



All other commands for this command set are described in this section in alphabetical order.



# aaa accounting commands *<level>*

Use **aaa accounting commands** to set parameters for authentication, authorization, and accounting (AAA) accounting. For more detailed information on AAA functionality, refer to the Technology Review section of the command *aaa on* on page 441. Use the **no** form of this command to disable these features. Variations of this command include:

**aaa accounting commands** *<level>* **default none**
**aaa accounting commands** *<level>* **default none group tacacs+**
**aaa accounting commands** *<level>* **default none group** *<groupname>*
**aaa accounting commands** *<level>* **default stop-only group tacacs+**
**aaa accounting commands** *<level>* **default stop-only group** *<groupname>*
**aaa accounting commands** *<level>* *<listname>* **none**
**aaa accounting commands** *<level>* *<listname>* **none group tacacs+**
**aaa accounting commands** *<level>* *<listname>* **none group** *<groupname>*
**aaa accounting commands** *<level>* *<listname>* **stop-only group tacacs+**
**aaa accounting commands** *<level>* *<listname>* **stop-only group** *<groupname>*

## Syntax Description

| | |
|---|---|
| *<level>* | Specifies the commands enable level. Only level 1 (unprivileged) and level 15 (privileged) commands are supported. |
| *<listname>* | Specifies the name of the list. |
| **default** | Uses the default accounting list. |
| **none** | Disables accounting. |
| **stop-only** | Records stop-only activity when service terminates. |
| **group** *<groupname>* | Uses the specified group of remote servers for accounting. |
| **group tacacs+** | Uses the TACACS+ server for accounting. |

## Default Values

By default, AAA accounting is disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Functional Notes

Use the **tacacs-server** command to specify TACACS+ servers before adding them to a group. This command enters a mode for adding individual servers to the named group. Refer to *TACACS+ Group Configuration Command Set* on page 2027 for more information.

The default group cannot be changed and includes all TACACS+ servers specified by the **tacacs-server** commands.

## Usage Examples

The following example creates a list called **myList** and sets accounting for level **1** commands at **stop-only** activities:

(config)**#aaa accounting commands 1 myList stop-only group tacacs+**


*To complete this command, Telnet must be applied to the lines. Refer to Line (Telnet) Interface Config Command Set* *on page 759* *for more detailed instructions.*

# aaa accounting connection

Use the **aaa accounting connection** command to send accounting records for outbound Telnet connections. For more detailed information on AAA functionality, refer to the *Technology Review* section of the command *aaa on* on page 441. Use the **no** form of this command to disable these features. Variations of this command include:

**aaa accounting connection** <*listname*> **none**
**aaa accounting connection** <*listname*> **start-stop group** <*groupname*>
**aaa accounting connection** <*listname*> **start-stop group tacacs+**
**aaa accounting connection** <*listname*> **stop-only group** <*groupname*>
**aaa accounting connection** <*listname*> **stop-only group tacacs+**
**aaa accounting connection default none**
**aaa accounting connection default start-stop group** <*groupname*>
**aaa accounting connection default start-stop group tacacs+**
**aaa accounting connection default stop-only group** <*groupname*>
**aaa accounting connection default stop-only group tacacs+**

## Syntax Description

| | |
|---|---|
| **default** | Uses the default accounting list. |
| **group** <*groupname*> | Specifies to use the named group remote server for accounting. Multiple groups can be specified. If the unit fails to make a connection with the first group, it will try the next group specified. |
| **group tacacs+** | Specifies to use the TACACS+ server for accounting. |
| <*listname*> | Specifies the name of the accounting list. |
| **none** | Disables accounting. |
| **start-stop** | Records when service begins and terminates. |
| **stop-only** | Records stop-only activity when service terminates. |

## Default Values

By default, AAA accounting connection is disabled.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Functional Notes

Use the **tacacs-server** command to specify TACACS+ servers before adding them to a group. This command enters a mode for adding individual servers to the named group. Refer to *TACACS+ Group Configuration Command Set* on page 2027 for more information.

The default group cannot be changed and includes all TACACS+ servers specified by the **tacacs-server** commands.

## Usage Examples

The following example creates a list called **myList** and sends the Telnet connection information to the TACACS+ server when the connection terminates:

(config)**#aaa accounting connection myList stop-only group tacacs+**

The following example creates a list called **myList** and sends the Telnet connection information to the TACACS+ server when the connection is made and when the connection terminates:

(config)**#aaa accounting connection myList start-stop group tacacs+**


*To complete these commands, Telnet must be applied to the lines. Refer to Line (Telnet) Interface Config Command Set* *on page 759* *for more detailed instructions.*

# aaa accounting exec

Use the **aaa accounting exec** command to send accounting records of all new connections/logins. For more detailed information on AAA functionality, refer to the *Technology Review* section of the command *aaa on* on page 441. Use the **no** form of this command to disable these features. Variations of this command include:

**aaa accounting exec** <*listname*> **none**
**aaa accounting exec** <*listname*> **start-stop group** <*groupname*>
**aaa accounting exec** <*listname*> **start-stop group tacacs+**
**aaa accounting exec** <*listname*> **stop-only group** <*groupname*>
**aaa accounting exec** <*listname*> **stop-only group tacacs+**
**aaa accounting exec default none**
**aaa accounting exec default start-stop group** <*groupname*>
**aaa accounting exec default start-stop group tacacs+**
**aaa accounting exec default stop-only group** <*groupname*>
**aaa accounting exec default stop-only group tacacs+**

## Syntax Description

| | |
|---|---|
| **default** | Uses the default accounting list. |
| **group** <*groupname*> | Specifies to use the named group remote server for accounting. Multiple groups can be specified. If the unit fails to make a connection with the first group, it will try the next group specified. |
| **group tacacs+** | Specifies to use the TACACS+ server for accounting. |
| <*listname*> | Specifies the name of the accounting list. |
| **none** | Disables accounting. |
| **start-stop** | Records when service begins and terminates. |
| **stop-only** | Records stop-only activity when service terminates. |

## Default Values

By default, AAA accounting exec is disabled.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Functional Notes

Use the **tacacs-server** command to specify TACACS+ servers before adding them to a group. This command enters a mode for adding individual servers to the named group. Refer to *TACACS+ Group Configuration Command Set* on page 2027 for more information.

The default group cannot be changed and includes all TACACS+ servers specified by the **tacacs-server** commands.

## Usage Examples

The following example creates a list called **myList** and sends the connection/login records to the TACACS+ server when the connection/login is terminated:

(config)**#aaa accounting exec myList stop-only group tacacs+**

*To complete this command, Telnet must be applied to the lines. Refer to Line (Telnet) Interface Config Command Set* *on page 759* *for more detailed instructions.*

# aaa accounting suppress null-username

Use the **aaa accounting suppress null-username** command to stop sending accounting records for usernames set to null. For more detailed information on authentication, authorization, and accounting (AAA) functionality, refer to the Technology Review section of the command *aaa on* on page 441. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled and the accounting records for null usernames are sent to the server.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following command tells the unit not to send accounting records for users with null usernames:

(config)**#aaa accounting suppress null-username**

# aaa accounting update

Use the **aaa accounting update** command to specify when accounting records are sent to the server. For more detailed information on authentication, authorization, and accounting (AAA) functionality, refer to the *Technology Review* section of the command *aaa on* on page 441. Use the **no** form of this command to return to the default settings. Variations of this command include:

**aaa accounting update newinfo**
**aaa accounting update periodic** <*value*>

## Syntax Description

| | |
|---|---|
| **newinfo** | Sends all new accounting records immediately. |
| **periodic** <*value*> | Specifies the time interval (in minutes) between accounting updates sent to the server. Select from 1 to 2,147,483,647. |

## Default Values

By default, accounting records are sent every 5 minutes.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following command sets the unit to send accounting records every **600** minutes to the server:

(config)**#aaa accounting update periodic 600**

# aaa authentication

Use the **aaa authentication** command to control various features of the AAA subsystem authentication process. For more detailed information on authentication, authorization, and accounting (AAA) functionality, refer to the *Technology Review* section of the command *aaa on* on page 441. Use the **no** form of this command to return to the default settings. Variations of this command include:

**aaa authentication banner** <*banner*>
**aaa authentication fail-message** <*message*>
**aaa authentication banner password-prompt** <*prompt*>
**aaa authentication banner username-prompt** <*prompt*>

## Syntax Description

| | |
|---|---|
| **banner** <*banner*> | Sets the banner shown before user authentication is attempted. The banner can be multiple lines. Enter a delimiter (such as :) to begin recording the typed text message used for the banner. The message must end with the same delimiter to indicate that the message is complete. The text delimiters are not displayed to the screen during operation. |
| **fail-message** <*message*> | Sets the message shown if user authentication fails. The message can be multiple lines. Enter a delimiter (such as :) to begin recording the typed text message displayed after a failed authentication attempt. The message must end with the same delimiter to indicate that the message is complete. The text delimiters are not displayed to the screen during operation. |
| **password-prompt** <*prompt*> | Sets the prompt for the user's password. The prompt is a single line enclosed in quotation marks. |
| **username-prompt** <*prompt*> | Sets the prompt for the user's name. The prompt is a single line enclosed in quotation marks. |

## Default Values

| | |
|---|---|
| **banner** | User Access Verification |
| **fail-message** | Authentication Failed |
| **password-prompt** | Password: |
| **username-prompt** | Username: |

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following is a typical example of customizing the AAA authentication process:

(config)**#aaa authentication banner #**
Enter TEXT message. End with the character '#'.
**User login authentication:#**
(config)#
(config)**#aaa authentication fail-message #**
Enter TEXT message. End with the character '#'.
**Authentication denied.#**
(config)#
(config)**#aaa authentication username-prompt “Enter Username:”**
(config)**#aaa authentication password-prompt “Enter Password:”**

# aaa authentication enable default

Use the **aaa authentication enable default** command to create (or change) the list of fallback methods used for privileged mode access authentication. For more detailed information on authentication, authorization, and accounting (AAA) functionality, refer to the *Technology Review* section of the command *aaa on* on page 441. Use the **no** form of this command to return to the default settings. Variations of this command include:

**aaa authentication enable default enable**
**aaa authentication enable default group radius**
**aaa authentication enable default group radius enable**
**aaa authentication enable default group tacacs+**
**aaa authentication enable default group tacacs+ enable**
**aaa authentication enable default group** <*name*>
**aaa authentication enable default group** <*name*> **enable**
**aaa authentication enable default line**
**aaa authentication enable default line enable**
**aaa authentication enable default none**
**aaa authentication enable default none enable**

## Syntax Description

| | |
|---|---|
| **none** | Grants access automatically. |
| **line** | Uses the line password for authentication. |
| **enable** | Uses the Enable mode password for authentication. |
| **group** <*name*> | Uses the specified group of remote servers for authentication. |
| **group radius** | Uses all defined RADIUS servers for authentication. |
| **group tacacs+** | Uses all defined TACACS+ servers for authentication. |

## Default Values

If no default methods list is configured, the unit uses the Enable password for authentication. If no password is configured, consoles are allowed access (this prevents a lock-out condition).

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |
| Release 11. | The **group tacacs+** command was added. |

## Functional Notes

A user is authenticated by trying the list of methods from first to last until authentication succeeds or fails. If a method does not succeed or fail, the next method is tried. The **group** methods will fail if the servers in the remote group cannot be found. Refer to the command *radius-server* on page 646 or *tacacs-server* on page 687 for information on defining server groups.

*Access to the Enable command set is a password-only process. The local-user database cannot be used, and the user name given to any remote RADIUS server is **$enab15$**. The only list name allowed is **default**.*

Use the **radius-server** command to specify RADIUS servers before adding them to a group. Likewise, use the **tacacs-server** command to specify TACACS+ servers before adding them to a group. These commands enter a mode for adding individual servers to the named group. Refer to *Radius Group Command Set* on page 1999 or *TACACS+ Group Configuration Command Set* on page 2027 for more information.

The default group cannot be changed and includes all RADIUS servers in the order they were specified by the **radius-server** commands. The same is true of TACACS+ servers specified by the **tacacs-server** commands.

## Usage Examples

The following example specifies using the **line** password as the first method of authentication and using the **Enable** password as the second:

(config)**#aaa authentication enable default line enable**

# aaa authentication login

Use the **aaa authentication login** command to create (or change) a named list with the ability to have a chain of fallback authentication methods for user authentication. For more detailed information on authentication, authorization, and accounting (AAA) functionality, refer to the *Technology Review* section of the command *aaa on* on page 441. Use the **no** form of this command to remove a configured login. Variations of this command include:

**aaa authentication login default enable**
**aaa authentication login default group radius**
**aaa authentication login default group radius enable**
**aaa authentication login default group tacacs+**
**aaa authentication login default group tacacs+ enable**
**aaa authentication login default group** *<groupname>*
**aaa authentication login default group** *<groupname>* **enable**
**aaa authentication login default line**
**aaa authentication login default line enable**
**aaa authentication login default local**
**aaa authentication login default local enable**
**aaa authentication login default none**
**aaa authentication login default none enable**
**aaa authentication login** *<listname>* **enable**
**aaa authentication login** *<listname>* **group radius**
**aaa authentication login** *<listname>* **group radius enable**
**aaa authentication login** *<listname>* **group tacacs+**
**aaa authentication login** *<listname>* **group tacacs+ enable**
**aaa authentication login** *<listname>* **group** *<groupname>*
**aaa authentication login** *<listname>* **group** *<groupname>* **enable**
**aaa authentication login** *<listname>* **line**
**aaa authentication login** *<listname>* **line enable**
**aaa authentication login** *<listname>* **local**
**aaa authentication login** *<listname>* **local enable**
**aaa authentication login** *<listname>* **none**
**aaa authentication login** *<listname>* **none enable**

## Syntax Description

| | |
|---|---|
| *<listname>* | Specifies a named login list. |
| **default** | Uses the default list for authentication when no other list is assigned. |
| **none** | Grants access automatically. |
| **line** | Uses the line password (Telnet 0 through 4 or console 0 through 1) for authentication. |
| **enable** | Uses the Enable password for authentication. |
| **local** | Uses the local-user database for authentication. |
| **group** *<groupname>* | Uses the specified group of remote servers for authentication. |
| **group radius** | Uses all defined RADIUS servers for authentication. |
| **group tacacs+** | Uses all defined TACACS+ servers for authentication. |

## Default Values

By default, the login list named **default** is the list used to authenticate users when no other list is assigned.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |
| Release 11. | The **group tacacs+** command was added. |

## Functional Notes

A user is authenticated by trying the list of methods from first to last until authentication succeeds or fails. If a method does not succeed or fail, the next method is tried. The local-user database method fails if the user name does not appear in the database. The **group** methods fail if the servers in the remote group cannot be found. Refer to the command *radius-server* on page 646 or *tacacs-server* on page 687 for information on defining server groups.

## Usage Examples

The following example creates a named list called **myList** and specifies using the **local** database as the first method, **myGroup** as the second method, and **line** password as the third method for login authentication:

(config)**#aaa authentication login myList local group myGroup line**

The following command sets the **default** authentication list for logins to use the **local** database as the first fallback method:

(config)**#aaa authentication login default local**

# aaa authentication port-auth default

Use the **aaa authentication port-auth default** command to create a default list for port authentication. For more detailed information on authentication, authorization, and accounting (AAA) functionality, refer to the *Technology Review* section of the command *aaa on* on page 441. Use the **no** form of this command to remove a configured list. Variations of this command include:

**aaa authentication port-auth default group radius**
**aaa authentication port-auth default group radius group**
**aaa authentication port-auth default group** *<name>*
**aaa authentication port-auth default group** *<name>* **group**
**aaa authentication port-auth default local**
**aaa authentication port-auth default local group**
**aaa authentication port-auth default none**
**aaa authentication port-auth default none group**

## Syntax Description

| | |
|---|---|
| **none** | Grants access automatically. |
| **local** | Specifies using the local user database for authentication. |
| **group** *<name>* | Specifies a group of remote servers to use for authentication. |
| **group radius** | Specifies using all defined RADIUS servers for authentication. |

## Default Values

By default, the login list named **default** is used to authenticate users when no other list is assigned.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example specifies that the **local** user database be used for port authentication:

(config)**#aaa authentication port-auth default local**

# aaa authorization

Use the **aaa authorization** to enable or disable authorization for configuration mode commands and for console mode. For more detailed information on authentication, authorization, and accounting (AAA) functionality, refer to the *Technology Review* section of the command *aaa on* on page 441. Use the **no** form of this command to return to the default setting. Variations of this command include:

**aaa authorization config-command**
**aaa authorization console**

## Syntax Description

| | |
|---|---|
| **config-command** | Enables authorization for configuration mode commands. Only level 1 (unprivileged) and level 15 (privileged) commands are supported. |
| **console** | Allows authorization to be applied to the console. |

## Default Values

By default, authorization for console is disabled. However, configuration mode commands are authorized by default.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example enables authorization of configuration mode commands:

(config)**#aaa authorization config-command**

The following example enables authorization of console commands:

(config)**#aaa authorization console**

# aaa authorization commands <*level*>

Use **aaa authorization commands** to create (or change) a list of methods for user authorization. For more detailed information on authentication, authorization, and accounting (AAA) functionality, refer to the *Technology Review* section of the command *aaa on* on page 441. Use the **no** form of this command to remove a configured list of authorization commands. Variations of this command include:

**aaa authorization commands** <*level*> **default group tacacs+**
**aaa authorization commands** <*level*> **default group** <*groupname*>
**aaa authorization commands** <*level*> **default if-authenticated**
**aaa authorization commands** <*level*> **default none**
**aaa authorization commands** <*level*> <*listname*> **group tacacs+**
**aaa authorization commands** <*level*> <*listname*> **group** <*groupname*>
**aaa authorization commands** <*level*> <*listname*> **if-authenticated**
**aaa authorization commands** <*level*> <*listname*> **none**

## Syntax Description

| | |
|---|---|
| <*level*> | Specifies the command's enable level. Only level 1 (unprivileged) and level 15 (privileged) commands are supported. |
| <*listname*> | Specifies the name of the authorization list. |
| **default** | Specifies the default authorization list and applies it implicitly across all lines. |
| **none** | Grants access automatically. |
| **if-authenticated** | Succeeds if user has authenticated. |
| **group** <*groupname*> | Uses the specified group of remote servers for authorization. |
| **group tacacs+** | Uses all defined TACACS+ servers for authorization. |

## Default Values

By default, the authorization list named **default** is the default list used to authorize commands when no other list is assigned.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following command creates a list called **myList** to authorize unprivileged commands (which succeeds only if the user has been authenticated successfully):

(config)**#aaa authorization commands 1 myList if-authenticated**

The following command uses the **default** list to authorize privileged (level 15) commands against the defined TACACS+ servers:

(config)**#aaa authorization commands 15 default group tacacs+**

# aaa authorization exec

Use the **aaa authorization exec** command to set authorization lists for exec shell. This allows a user to enter directly into Enable mode for new CLI sessions when authorized. For more detailed information on authentication, authorization, and accounting (AAA) functionality, refer to the *Technology Review* section of the command *aaa on* on page 441. Use the **no** form of this command to remove a configured list of authorization commands. Variations of this command include:

**aaa authorization exec** <*listname*> **none**
**aaa authorization exec** <*listname*> **group** <*groupname*>
**aaa authorization exec** <*listname*> **group tacacs+**
**aaa authorization exec** <*listname*> **if-authenticated**
**aaa authorization exec** <*listname*> **if-authenticated group** <*groupname*>
**aaa authorization exec** <*listname*> **if-authenticated group tacacs+**
**aaa authorization exec default none**
**aaa authorization exec default group** <*groupname*>
**aaa authorization exec default group tacacs+**
**aaa authorization exec default if-authenticated**
**aaa authorization exec default if-authenticated group** <*groupname*>
**aaa authorization exec default if-authenticated group tacacs+**

## Syntax Description

| | |
|---|---|
| **default** | Specifies the default authorization list and applies it implicitly across all lines. |
| **group** <*groupname* | Specifies to use the named group remote server or the TACACS+ server for authorization. Multiple groups can be specified. If the unit fails to make a connection with the first group, it will try the next group specified. However, if a connection is made to the first group and authorization fails, the unit will not attempt authorization with any other groups in the list. |
| **tacacs+** | Specifies to use the TACACS+ server for authorization. |
| **if-authenticated** | Succeeds if user has authenticated. |
| <*listname*> | Specifies the name of the authorization list. |
| **none** | Grants access automatically. |

## Default Values

By default, the authorization exec is disabled.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following command creates a list called **myList** to authorize exec shell (which succeeds only if the user has been authenticated successfully):

(config)**#aaa authorization exec myList if-authenticated**

The following command specifies to use the default list to authorize an exec shell with the TACACS+ server:

(config)**#aaa authorization exec default group tacacs+**

# aaa group server

Use the **aaa group server** command to group pre-defined RADIUS and TACACS+ servers into named lists. For more detailed information on authentication, authorization, and accounting (AAA) functionality, refer to the *Technology Review* section of the command *aaa on* on page 441. Use the **no** form of this command to remove a configured server group. Variations of this command include:

**aaa group server radius** *<name>*
**aaa group server tacacs+** *<name>*

## Syntax Description

| | |
|---|---|
| **radius** | Groups defined RADIUS servers. |
| **tacacs+** | Groups defined TACACS+ servers. |
| *<name>* | Specifies the name of the server list. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |
| Release 11.1 | Command was expanded to include TACACS+ server support. |

## Functional Notes

Use the **radius-server** command to specify RADIUS servers before adding them to a group. Likewise, use the **tacacs-server** command to specify TACACS+ servers before adding them to a group. These commands enter a mode for adding individual servers to the named group. Refer to *Radius Group Command Set* on page 1999 or *TACACS+ Group Configuration Command Set* on page 2027 for more information.

The default group cannot be changed and includes all RADIUS servers in the order they were specified by the **radius-server** commands. The same is true of TACACS+ servers specified by the **tacacs-server** commands.

## Usage Examples

The following example creates the named list **myServers** and enters the RADIUS group:

```
(config)#aaa group server radius myServers
(config-sg-radius)#
```

The following example creates the named list **myServers** and enters the TACACS+ group:

```
(config)#aaa group server tacacs myServers
(config-sg-tacacs+)#
```

# aaa on

Use the **aaa** on command to activate the authentication, authorization, and accounting (AAA) subsystem. Use the **no** form of this command to deactivate AAA.

## Syntax Description

No subcommands.

## Default Values

By default, AAA is not activated.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

By default, the AAA subsystem is disabled and authentication follows the line technique (local, line, etc.). Once activated, the AAA lists override the methods specified in the line command.

## Usage Examples

The following example activates the AAA subsystem:

(config)**#aaa on**

## Technology Review

AAA stands for authentication, authorization, and accounting. The AOS AAA subsystem currently supports only authentication. Authentication is the means by which a user is granted access to the device (router). For instance, a user name/password is authenticated before the user can use the CLI. VPN clients can also verify user name/password before getting access through the device.

There are several methods that can be used to authenticate a user:

| | |
|---|---|
| **none** | Grants instant access. |
| **line** | Uses the line password (Telnet 0 through 4 or console 0 through 1). |
| **enable** | Uses the Enable password. |
| **local** | Uses the local-user database. |
| **group** <*name*> | Uses the specified group of remote servers for authentication. |
| **group radius** | Uses all defined RADIUS servers for authentication. |
| **group tacacs+** | Uses all defined TACACS+ servers for authentication. |

The AAA system allows users to create a named list of authentication methods to attempt in order (if one fails, it falls to the next one on the list). This named list is then attached to a portal (Telnet 0 through 4 or console 0 through 1). When a user Telnets in or accesses the terminal, the AAA system uses the methods from the named list to authenticate the user.

By default the AAA system is disabled. It must be turned on to be active. Use the **aaa on** command to activate the AAA system.

If a portal is not explicitly assigned a named list, the name **default** is automatically assigned to it. Users can customize the **default** list just like any other list. If no **default** list is configured, the following default behavior applies (defaults are based on portal):

- Instant access (**none**) is assigned to the console using the **default** list (when the list has not been configured).
- The local-user database (**local**) is used for Telnet sessions using the **default** list (when the list has not been configured).
- No FTP access is granted using the **default** list (when the list has not been configured).

Methods fail (and therefore cause the system to proceed to the next configured method) under the following circumstances:

- **line** and **enable** fail if there are no line or Enable passwords configured.
- **local** fails if the given user does not appear in the local-user database.
- **group** fails if the given server(s) cannot be contacted on the network.

**Example**

For a **default** list defined with the order [**line**, **enable**, **local**, **group mygroup**], the following statements are true:

- If there is no line password, the list falls through to the Enable password.
- If there is no Enable password, the AAA system prompts the user for a user name and password for the local-user database.
- If the given user is not in the local-user database, the user name and password are handed to the remote servers defined in **mygroup**.
- A failure at any point (password not matching) denies access.

If the AAA authentication process fails the list completely, system behavior is based on portal:

- Console access is granted if the process fails completely (this prevents a lock-out condition).
- Telnet and FTP are denied access.

# aaa processes *<value>*

Use the **aaa processes** command to set the number of threads available to the authentication, authorization, and accounting (AAA) subsystem. For more detailed information on AAA functionality, refer to the *Technology Review* section of the command *aaa on* on page 441. Use the **no** form of this command to return to the default setting.

## Syntax Description

| | |
|---|---|
| *<value>* | Specifies the number of threads available to the AAA subsystem. Range is 1 to 64 threads. |

## Default Values

By default, the number of threads is set to 1.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

Increasing the number of threads may speed up simultaneous authentication at the cost of system resources (e.g., memory).

## Usage Examples

The following example specifies five available threads for the AAA subsystem:

(config)**#aaa processes 5**

# **arp** *<ip address> <mac address>* **arpa**

Use the **arp arpa** command to enter static entries into the address resolution protocol (ARP) table. Use the **no** form of this command to remove a static ARP entry.

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies a valid IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| *<mac address>* | Specifies a valid 48-bit MAC address. MAC addresses should be expressed in following format xx:xx:xx:xx:xx:xx (for example, **00:A0:C8:00:00:01**). |

## Default Values

No default is necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following example enables standard ARP for the VLAN interface:

(config)**#interface vlan 1**
(config-interface-vlan 1)**#arp 196.173.22.253 00:A0:C8:00:00:01 arpa**

# auto-config

Use the **auto-config** command to enable the automatic self-configuration feature in AOS. For more detailed information on auto-config, refer to the *Auto-Config Configuration Guide* on the documentation CD, PN 61200560L1-29.2. Use the **no** form of this command to halt the auto-config process. Variations of this command include:

**auto-config**
**auto-config filename** <*name*>
**auto-config restart**
**auto-config retry-count** <*number*>
**auto-config server [**<*hostname*> **|** <*ip address*>**]**

> NOTE *Refer to the **Auto-Config Configuration Guide** (61200560L1-29.2) for more information on this command. This document is located on the ADTRAN OS System Documentation CD provided with your unit*

## Syntax Description

| | |
|---|---|
| **filename** <*name*> | Optional. Specifies the configuration filename to download. |
| **restart** | Optional. Restarts auto-config parameters. |
| **retry-count** <*number*> | Optional. Specifies the maximum number of retries. Range is 0 to 1000. |
| **server [**<*hostname*> **|** <*ip address*>**]** | Optional. Specifies the IP address or host name of TFTP Server from which to download. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

## Default Values

By default, auto-config is disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example enables auto-config:

(config)**#auto-config**

> NOTE *Disabling and re-enabling **auto-config** restarts the download process.*

The following command specifies the name of the file to download:

(config)**#auto-config filename myConfig**

The following command restarts the auto-config process:

(config)**#auto-config restart**

The following command sets the number of retries when downloading a configuration file to 100:

(config)**#auto-config retry-count 100**

The following command specifies the TFTP server IP address from which to download the configuration file:

(config)**#auto-config server 192.33.5.99**

The following command specifies the TFTP server host name from which to download the configuration file:

(config)**#auto-config server myHost**

# banner

Use the **banner** command to specify messages to be displayed in certain situations. Use the **no** form of this command to delete a previously configured banner. Variations of this command include:

**banner exec** *<delimiter> <message> <delimiter>*
**banner login** *<delimiter> <message> <delimiter>*
**banner motd** *<delimiter> <message> <delimiter>*

## Syntax Description

| | |
|---|---|
| **exec** | Creates a message to be displayed when any executive-level process takes place. |
| **login** | Creates a message to be displayed before the user name and password login prompts. |
| **motd** | Creates a message-of-the-day (MOTD) banner. |
| *<delimiter>* | Specifies the banner text delimiter. Press **Enter** after the delimiter character to begin input of banner text. After typing the banner message, enter the same delimiter character to end the message. |
| *<message>* | Specifies the text message you wish to display. |

## Default Values

By default, no banners are configured.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |

## Functional Notes

Banners appear in the following order (if configured):

- MOTD banner appears at initial connection.
- Login banner follows the MOTD banner.
- Exec banner appears after successful login.

## Usage Examples

The following example configures the system to display a message of the day:

(config)**#banner motd *The system will be shut down today from 7PM to 11PM***

# boot system

Use the **boot system** command to specify the system image loaded at startup.Variations of this command include:

**boot system cflash** *<primary filename>*
**boot system cflash** *<primary filename>* **verify**
**boot system flash** *<primary filename> <secondary filename>*
**boot system flash** *<primary filename> <secondary filename>* **verify**
**boot system cflash** *<primary filename>* **cflash** *<secondary filename>*
**boot system cflash** *<primary filename>* **cflash** *<secondary filename>* **verify**
**boot system cflash** *<primary filename>* **flash** *<secondary filename>*
**boot system cflash** *<primary filename>* **flash** *<secondary filename>* **verify**
**boot system cflash** *<primary filename>* **no-backup**
**boot system cflash** *<primary filename>* **no-backup verify**
**boot system flash** *<primary filename>* **cflash** *<secondary filename>*
**boot system flash** *<primary filename>* **cflash** *<secondary filename>* **verify**
**boot system flash** *<primary filename>* **flash** *<secondary filename>*
**boot system flash** *<primary filename>* **flash** *<secondary filename>* **verify**
**boot system flash** *<primary filename>* **no-backup**
**boot system flash** *<primary filename>* **no-backup verify**


*The **cflash** option is only valid for units with compact flash capabilities.*


*For units without compact flash capabilities, the secondary media type does not need to be specified. See the last example under Usage Examples.*

## Syntax Description

| | |
|---|---|
| **cflash** | Specifies the image located in compact flash memory. |
| **flash** | Specifies the system image loaded at startup. |
| **no-backup** | Specifies that there is no backup image present. |
| *<primary filename>* | Specifies the filename (located in flash memory) of the image (filenames are case-sensitive). Image files should have a .biz extension |
| *<secondary filename>* | Specifies a name for the backup image. |
| **verify** | Optional. Verifies the image checksum. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 12.1 | Command was expanded to include compact flash. |

## Functional Notes

Detailed instructions for upgrading the AOS and loading files into flash memory are found on the *ADTRAN OS System Documentation* CD.

## Usage Example

The following example specifies **myimage.biz** as the primary image file:

(config)**#boot system cflash myimage.biz no-backup**

The following example specifies **myimage.biz** as the primary image file with no backup image:

(config)**#boot system flash myimage.biz no-backup**

The following example specifies **myimage.biz** as the primary image file and **myimage_backup.biz** as the backup image:

(config)**#boot system flash myimage.biz myimage_backup.biz**

# boot voip

Use the **boot voip** command to specify the VoIP image file loaded at startup. Variations of this command include:

**boot voip default**
**boot voip flash** <*filename*>

## Syntax Description

| | |
|---|---|
| **default** | Uses default VoIP image. |
| **flash** <*filename*> | Specifies the filename (located in flash memory) of the image (file names are case-sensitive). Image files should have a .biz extension. |

## Default Values

No default value necessary for this command.

## Functional Notes

Detailed instructions for upgrading the AOS and loading files into flash memory are found on the *ADTRAN OS System Documentation* CD.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example specifies the file **myimage.biz**, stored in flash memory, as the VoIP startup image:

(config)**#boot voip flash myimage.biz**

# bridge <*number*> protocol ieee

The **bridge protocol ieee** command configures a bridge group for the IEEE 802.1 Ethernet spanning-tree protocol. Use the **no** form of this command (with the appropriate arguments) to delete this setting.

## Syntax Description

| | |
|---|---|
| <*number*> | Specifies a bridge group number. Range is 1 to 255. |

## Default Values

By default, all configured bridge interfaces implement **ieee** spanning-tree protocol.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example deletes the bridge protocol setting for bridge group **17**:

(config)**#no bridge 17 protocol ieee**

# clock

The **clock auto-correct-DST** command allows the unit to automatically correct for Daylight Saving Time (DST). Use the **clock no-auto-correct-DST** command to disable this feature. Variations of this command include:

**clock auto-correct-DST**
**clock no-auto-correct-DST**

## Syntax Description

| | |
|---|---|
| **auto-correct-DST** | Configures the unit to automatically correct for DST. |
| **no-auto-correct-DST** | Disables DST correction. |

## Default Values

By default, DST correction takes place automatically.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |
| Release 11.1 | Command was added to the Global command set. |

## Functional Notes

Depending on the **clock timezone** chosen (refer to *clock timezone* <*value*> on page 454 for more information) one-hour DST correction may be enabled automatically. You may override this default using this command.

## Usage Examples

The following example allows for automatic DST correction:

(config)**#clock auto-correct-dst**

The following example overrides the one-hour offset for DST:

(config)**#clock no-auto-correct-dst**

# clock set *<time> <day> <month> <year>*

Use the **clock set** command to configure the system software clock. For the command to be valid, all fields must be entered. Refer to the *Usage Examples* below for an example.

## Syntax Description

| | |
|---|---|
| *<time>* | Sets the time (in 24-hour format) of the system software clock in the format HH:MM:SS (hours:minutes:seconds). |
| *<day>* | Sets the current day of the month. Valid range is 1 to 31. |
| *<month>* | Sets the current month. Valid range is January to December. You need only enter enough characters to make the entry unique. This entry is not case-sensitive. |
| *<year>* | Sets the current year. Valid range is 2000 to 2100. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |
| Release 11.1 | Command was added to the Global command set. |

## Usage Examples

The following example sets the system software clock for 3:42 pm, August 22 2004:

(config)**#clock set 15:42:00 22 Au 2004**

# clock timezone <*value*>

The **clock timezone** command sets the unit's internal clock to the time zone of your choice. This setting is based on the difference in time (in hours) between Greenwich Mean Time (GMT) or Central Standard Time (CST) and the time zone for which you are setting up the unit. Use the **no** form of this command to disable this feature.

## Syntax Description

| | |
|---|---|
| <*value*> | Time zone values are specified in the *Functional Notes* section for this command. |

## Default Values

No default value is necessary for this command.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |


*Depending on the **clock timezone** chosen, one-hour Daylight Savings Time (DST) correction may be enabled automatically. Refer to clock* *on page 452* *for more information.*

## Functional Notes

The following list shows sample cities and their timezone codes.

clock timezone +1-Amsterdam
clock timezone +1-Belgrade
clock timezone +1-Brussels
clock timezone +1-Sarajevo
clock timezone +1-West-Africa
clock timezone +10-Brisbane
clock timezone +10-Canberra
clock timezone +10-Guam
clock timezone +10-Hobart
clock timezone +10-Vladivostok
clock timezone +11
clock timezone +12-Auckland
clock timezone +12-Fiji
clock timezone +13
clock timezone +2-Athens
clock timezone +2-Bucharest
clock timezone +2-Cairo
clock timezone +2-Harare
clock timezone +2-Helsinki
clock timezone +2-Jerusalem
clock timezone +3-Baghdad
clock timezone +3-Kuwait
clock timezone +3-Moscow
clock timezone +3-Nairobi
clock timezone +3:30
clock timezone +4-Abu-Dhabi
clock timezone +4-Baku
clock timezone +4:30
clock timezone +5-Ekaterinburg
clock timezone +5-Islamabad
clock timezone +5:30
clock timezone +5:45
clock timezone +6-Almaty
clock timezone +6-Astana
clock timezone +6-Sri-Jay
clock timezone +6:30
clock timezone +7-Bangkok
clock timezone +7-Kranoyarsk
clock timezone +8-Bejing
clock timezone +8-Irkutsk
clock timezone +8-Kuala-Lumpur
clock timezone +8-Perth
clock timezone +8-Taipei
clock timezone +9-Osaka
clock timezone +9-Seoul
clock timezone +9-Yakutsk
clock timezone +9:30-Adelaide
clock timezone +9:30-Darwin
clock timezone -1-Azores
clock timezone -1-Cape-Verde
clock timezone -10
clock timezone -11
clock timezone -12
clock timezone -2
clock timezone -3-Brasilia
clock timezone -3-Buenos-Aires
clock timezone -3-Greenland
clock timezone -3:30
clock timezone -4-Atlantic-Time
clock timezone -4-Caracus
clock timezone -4-Santiago
clock timezone -5
clock timezone -5-Bogota
clock timezone -5-Eastern-Time
clock timezone -6-Central-America
clock timezone -6-Central-Time
clock timezone -6-Mexico-City
clock timezone -6-Saskatchewan
clock timezone -7-Arizona
clock timezone -7-Mountain-Time
clock timezone -8
clock timezone -9
clock timezone -0-Universal Coordinated Time (UTC)
clock timezone GMT-Casablanca
clock timezone GMT-Dublin

## Usage Examples

The following example sets the timezone for Santiago, Chile.

>**enable**
(config)**#clock timezone -4-Santiago**

# cross-connect

Use the **cross-connect** command to create a cross-connect map from a created TDM group on an interface to a virtual interface. Variations of this command include:

**cross-connect** *<number> <from interface> <to interface>*
**cross-connect** *<number> <from interface> <group number> <to interface>*


*Changing **cross-connect** settings could potentially result in service interruption.*

## Syntax Description

| | |
|---|---|
| *<number>* | Identifies the cross-connect using a number descriptor or label for (useful in systems that allow multiple cross-connects). |
| *<from interface>* | Specifies the interface (physical or virtual) on one end of the cross-connect. Specify an interface in the format *<interface type [slot/port \| slot/port.sub-interface id \| interface id \| interface id.sub-interface id]>*. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Enter **cross-connect 1 ?** for a list of valid interfaces. |
| *<group number>* | Optional. Specifies which configured TDM group to use for this cross-connect. This subcommand only applies to T1 physical interfaces. |
| *<to interface>* | Specifies the virtual interface on the other end of the cross-connect. Specify an interface in the format *<interface type [slot/port \| slot/port.sub-interface id \| interface id \| interface id.sub-interface id]>*. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Use the **?** to display a list of valid interfaces. |

## Default Values

By default, there are no configured cross-connects.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include the E1 interface. |

## Functional Notes

Cross-connects provide the mechanism for connecting a configured virtual (layer 2) endpoint with a physical (layer 1) interface. Supported layer 2 protocols include Frame Relay and point-to-point protocol (PPP).

## Usage Examples

The following example creates a Frame Relay endpoint and connects it to the T1 1/1 physical interface:

1. Create the Frame Relay virtual endpoint and set the signaling method:
   (config)**#interface frame-relay 1**
   (config-fr 1)**#frame-relay lmi-type cisco**

2. Create the sub-interface and configure the PVC parameters (including DLCI and IP address):
   (config-fr 1)**#interface fr 1.1**
   (config-fr 1.1)**#frame-relay interface-dlci 17**
   (config-fr 1.1)**#ip address 168.125.33.252 255.255.255.252**

3. Create the TDM group of 12 DS0s (64K) on the T1 physical interface:
   (THIS STEP IS ONLY VALID FOR T1 INTERFACES.)
   (config)**#interface t1 1/1**
   (config-t1 1/1)**#tdm-group 1 timeslots 1-12 speed 64**
   (config-t1 1/1)**#exit**

4. Connect the Frame Relay sub-interface with port T1 1/1:
   (config)**#cross-connect 1 t1 1/1 1 fr 1**

## Technology Review

Creating an endpoint that uses a layer 2 protocol (such as Frame Relay) is generally a four-step process:

Step 1:
Create the Frame Relay virtual endpoint (using the **interface frame-relay** command) and set the signaling method (using the **frame-relay lmi-type** command). Also included in the Frame Relay virtual endpoint are all the applicable Frame Relay timers logging thresholds, encapsulation types, etc. Generally, most Frame Relay virtual interface parameters should be left at their default state. For example, the following creates a Frame Relay interface labeled **7** and sets the signaling method to **ansi**.

(config)**#interface frame-relay 7**
(config-fr 7)**#frame-relay lmi-type ansi**

Step 2:
Create the sub-interface and configure the PVC parameters. Using the sub-interface, apply access policies to the interface, create bridging interfaces, configure dial-backup, assign an IP address, and set the PVC data-link control identifier (DLCI). For example, the following creates a Frame Relay sub-interface labeled **22,** sets the DLCI to **30**, and assigns an IP address of **193.44.69.253** to the interface.

(config-fr 7)**#interface fr 7.22**
(config-fr 7.22)**#frame-relay interface-dlci 30**
(config-fr 7.22)**#ip address 193.44.69.253 255.255.255.252**

Step 3: (VALID ONLY FOR T1 INTERFACES)
Specify the group of DS0s used for signaling on the T1 interface by creating a TDM group. Group any number of contiguous DS0s together to create a data pipe for layer 2 signaling. Also use the **tdm-group** command to specify the per-DS0 signaling rate on the interface. For example, the following creates a TDM group labeled **9** containing 20 DS0s (each DS0 having a data rate of 56 kbps).

(config)**#interface t1 1/1**
(config-t1 1/1)**#tdm-group 9 timeslots 1-20 speed 56**
(config-t1 1/1)**#exit**

Step 4:
Make the association between the layer 2 endpoint and the physical interface using the **cross-connect** command. Supported layer 2 protocols include Frame Relay and point-to-point protocol (PPP). For example, the following creates a cross-connect (labeled **5**) to make an association between the Frame Relay virtual interface (**fr 7**) and the TDM group configured on interface t1 1/1 (**tdm-group 9**).

(config)**#cross-connect 5 t1 1/1 9 fr 7**

# crypto ca authenticate *<name>*

Use the **crypto ca authenticate** command to initiate CA authentication procedures.

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies a CA profile using an alphanumeric string up to 32 characters. |

## Default Values

No defaults necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

The type of authentication procedure is based on the **enrollment** command and its settings. Refer to *enrollment terminal* on page 1590 and *enrollment url <url>* on page 1591 for more information. When **enrollment** is set to **terminal**, the CA authentication process is done manually, as shown in the following *Usage Examples*.

## Usage Examples

The following example initiates the CA authentication process:

(config)**#crypto ca authenticate testCAprofile**

Enter the base 64 encoded CA certificate. End with two consecutive carriage returns or the word “quit” on a line by itself:

-----BEGIN X509 CERTIFICATE-----
MIIDEDCCAs6gAwIBAgICAXIwCwYHKoZIzjgEAwUAMFoxCzAJBgNVBAYTAkZJMSQw
IgYDVQQKExtTU0ggQ29tbXVuaWNhdGlvbnMgU2VjdXJpdHkxETAPBgNVBAsTCFdl
YiB0ZXN0MRIwEAYDVQQDEwlUZXN0IENBIDQwHhcNMDMwMTA5MTYyNTE1WhcNMDMx
MjMxMjM1OTU5WjBaMQswCQYDVQQGEwJGSTEkMCIGA1UEChMbU1NIIENvbW11bmlj
YXRpb25zIFNlY3VyaXR5MREwDwYDVQQLEwhXZWIgdGVzdDESMBAGA1UEAxMJVGVz
dCBDQSA0MIIBtzCCASsGByqGSM44BAEwggEeAoGBAPTo+NdCWh87hOSnuZ7dUL07
twjZZwY3beLHnDsERhfN8XoOZZcffulKc/lqTrYiu7M5yPJsXQ3u8dbCb6RWFU0A
T5Nd7/4cNn/hCmhbeb6xqsNZUsOcTZJxvClq8thkNo+gXg5bw0fiElgxZ/lEbFWL
UzeO8KgM4izkq0CrGtaFAhUA2+ja4RgbbgTgJk+qTXAxicG/8JMCgYBZvcPMO2/Y
Zc2sXYyrBPtv6k2ZGGYqXAUZ98/txm37JwQGafygePJ/64oeisVeDcLf2FTjveex
W5saydjSK00jXjreRZcJFEDmfRhUtWR8K8tm8mEnB3eg9n09lkWibIjihHn7n5MF
tBBAdbRHyctsr3DyofnieTt3DY78MDsNbgOBhQACgYEA6EKDS2IxrdMsogHfVvob
PkDSv2FjOsP5Tomc/tf9jvvuf6+vj9XTw+uAg1BU9/TyjGzAtnRrCvOUkTYoVxRY
vdDOi3GR2RcyNVdGrhYXWY1I5XuB5+NWij8VUQOgfXsJgbEMvPemECeYwQ4ASdhD
vw0E8NI2AEkJXsCAvYfXWzujIzAhMAsGA1UdDwQEAwIBhjASBgNVHRMBAf8ECDAG
AQH/AgEyMAsGByqGSM44BAMFAAMvADAsAhRa0ao0FbRQeWCc2oC24OZ1YZi8egIU

lZhxKAclhXksZHvOj+yIld5x0ec=
-----END X509 CERTIFICATE-----
**quit**
Hash: 4e904504dc4e5b95e08129430e2a0b97ceef0ad1394f905b42df2dfb8f751be0244a711bb0
6eddaa2f07dd640c187f14c16fa0bed28e038b28b6741a880539d6ed06a68b7e324bfdde6f3d0b17
83d94e58fd4943f5988a7a0f27f6b6b932dc041037824716075285385 8dbe7a1951245cfb14b109e
ffc430e177623720de56f4
* Do you accept this certificate? [y]**y**

# crypto ca certificate chain <*name*>

Use the **crypto ca certificate chain** command to enter the Certificate Configuration for the specified CA. Refer to *Certificate Configuration Command Set* on page 1597 for more information.

## Syntax Description

| | |
|---|---|
| <*name*> | Specifies a CA profile using an alphanumeric string (up to 32 characters). |

## Default Values

No defaults necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

Typically used only in the **running-config** and **startup-config** to restore certificates.

## Usage Examples

The following example enters the Certificate Configuration mode for the CA profile **MyProfile**:

(config)**#crypto ca certificate chain MyProfile**

# crypto ca enroll <*name*>

Use the **crypto ca enroll** command to begin CA enrollment procedures. Use the **no** form of this command to disable this feature.

## Syntax Description

| | |
|---|---|
| <*name*> | Specifies a CA profile using an alphanumeric string (up to 32 characters). |

## Default Values

No defaults necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

The type of enrollment procedure is based on the **enrollment** command and its settings. Refer to *enrollment terminal* on page 1590 and *enrollment url <url>* on page 1591 for more information. This command initiates a dialog that is used to fill in the parameters that make up an enrollment request to be forwarded to a certificate authority. Note that some of the parameters (such as IP address) may be filled in using the values supplied in the **crypto ca profile** (in which case, the enrollment dialog will not prompt for those parameters). Once all required parameters are defined using the dialog, this command assembles them into an enrollment request to be sent to a certificate authority (including the generation of public and private keys). Refer to *crypto ca profile <name>* on page 466 for more information.

If **enrollment** is set to **terminal**, you may view the request on the terminal screen.

If **enrollment** is set to **url**, the request is sent automatically to the certificate authority using the URL specified by the **enrollment url** command.

## Usage Examples

The following example shows a typical enrollment dialog:

(config)**#crypto ca enroll MyProfile**
**** Press CTRL+C to exit enrollment request dialog. ****
* Enter signature algorithm (RSA or DSS) [rsa]:**rsa**
* Enter the modulus length to use [512]:**1024**
* Enter the subject name as an X.500 (LDAP) DN:**CN=Router,C=US,L=Huntsville,S=AL**
  --The subject name in the certificate will be CN=CN=Router,C=US,L=Huntsville,S=AL.
* Include an IP address in the subject name [n]:**y**
* Enter IP address or name of interface to use:**10.200.1.45**
* Include fully qualified domain name [n]:**y**
* Enter the fully qualified domain name to use:**FullyQualifiedDomainName**
* Include an email address [n]:**y**
* Enter the email address to use:**myEmail@adtran.commyemail@email.com**
Generating request (including keys)....

# crypto ca import <*name*> certificate

Use the **crypto ca import certificate** command to import a certificate manually via the console terminal.

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies a CA profile using an alphanumeric string (up to 32 characters). |

## Default Values

No defaults necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

Puts CLI in mode where the certificate can be entered manually. Enter **quit** and a carriage return (or simply enter two consecutive carriage returns) to exit this mode. Abort this mode by pressing **Ctrl-C**. This command only applies if the **enrollment** command is set to **terminal**. Refer to *enrollment terminal* on page 1590.

## Usage Examples

The following example imports a certificate via the console terminal:

```
(config)#crypto ca import MyProfile certificate
Enter the PM-encoded certificate. End with two consecutive
carriage returns or the word “quit” on a line by itself:
-----BEGIN CERTIFICATE-----
MIIDWTCCAwOgAwIBAgIKFLCsOgAAAAAtjANBgkqhkiG9w0BAQUFADBjMQswCQYD
VQQGEwJVUzEQMA4GA1UECBMHQUxBQkFNQTETMBEGA1UEBxMKSHVudHN2aWxsZTEa
MBgGA1UEChMRQWR0cmFuVGVjaFN1cHBvcnQxETAPBgNVBAMTCHRzcm91dGVyMB4X
DTAzMDYyNTE0MTM1NVoXDTAzMTIwNjE0NDkxM1owJDEPMA0GA1UEChMGYWR0cmFu
MREwDwYDVQQDEwhNeVJvdXRlcjBcMA0GCSqGSIb3DQEBAQUAA0sAMEgCQQClUKqs
fbTalej5m9gk2DMsbC9df3TilBz+7nRx3ZzGw75AQsqEMYeBY5aWi62W59jmxGSE
WX+E8EwBVbZ6JKk5AgMBAAGjggHWMIIB0jAXBgNVHREEEDAOhwQKCgoKggZNeUZx
ZG4wHQYDVR0OBBYEFJAvBRIjx1PROnkZ4v0D89yB1eErMIGcBgNVHSMEgZQwgZGA
FHGwIRAr11495MgrLNPiLzjvrb4JoWekZTBjMQswCQYDVQQGEwJVUzEQMA4GA1UE
CBMHQUxBQkFNQTETMBEGA1UEBxMKSHVudHN2aWxsZTEaMBgGA1UEChMRQWR0cmFu
VGVjaFN1cHBvcnQxETAPBgNVBAMTCHRzcm91dGVyghAZql7OwISgsUhfaSeGh0Ot
MGkGA1UdHwRiMGAwLaAroCmGJ2h0dHA6Ly90c3JvdXRlci9DZXJ0RW5yb2xsL3Rz
cm91dGVyLmNybDAvoC2gK4YpZmlsZTovL1xcdHNyb3V0ZXJcQ2VydEVucm9sbFx0
c3JvdXRlci5jcmwwgY0GCCsGAQUFBwEBBIGAMH4wPAYIKwYBBQUHMAKGMGh0dHA6
Ly90c3JvdXRlci9DZXJ0RW5yb2xsL3Rzcm91dGVyX3Rzcm91dGVyLmNydDA+Bggr
BgEFBQcwAoYyZmlsZTovL1xcdHNyb3V0ZXJcQ2VydEVucm9sbFx0c3JvdXRlcl90
-----END CERTIFICATE-----
Success!
```

# crypto ca import <*name*> crl

Use the **crypto ca import crl** command to import a CRL manually via the console terminal.

## Syntax Description

| | |
|---|---|
| <*name*> | Specifies a CA profile using an alphanumeric string (up to 32 characters). |

## Default Values

No defaults necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

Puts CLI in a mode where the CRL can be entered manually. Enter **quit** and a carriage return (or simply enter two consecutive carriage returns) to exit this mode. This command only applies if the **enrollment** command is set to **terminal**. Refer to *enrollment terminal* on page 1590.

## Usage Examples

The following allows you to manually paste in the CA's CRL:

(config)**#crypto ca import MyProfile crl**

# crypto ca profile *<name>*

Use the **crypto ca profile** command to define a CA and to enter the CA Profile Configuration. Use the **no** form of this command to disable this feature. Refer to *CA Profile Configuration Command Set* on page 1586 for more information.

## Syntax Description

| | |
|---|---|
| *<name>* | Creates a CA profile using an alphanumeric string (up to 32 characters). |

## Default Values

No defaults necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

Use this to specify the type of enrollment, as well as enrollment request parameters. Refer to the *Functional Notes* of the command *crypto ca enroll <name>* on page 463 for more information.

## Usage Examples

The following example creates the CA profile called **MyProfile** and enters the CA Profile Configuration for that certificate authority:

```
(config)#crypto ca profile MyProfile
Configuring New CA Profile MyProfile.
(ca-profile)#
```

# crypto ike

Use the **crypto ike** command to define the system-level local ID for IKE negotiations and to enter the IKE Client or IKE Policy command sets. Variations of this command include the following: Use the **no** form of this command to disable these features.

**crypto ike client configuration pool** <*name*>
**crypto ike local-id address**
**crypto ike policy** <*value*>

## Syntax Description

| | |
|---|---|
| **client configuration pool** <*name*> | Creates a local pool, assigns it the name of your choice and enters the IKE Client command set. Clients that connect via an IKE policy that specifies this pool name will be assigned values from this pool. Refer to the section *IKE Policy Command Set* on page 1633 for more information. |
| **local-id address** | Sets the local ID during IKE negotiation to be the IP address of the interface from which the traffic exits. This setting can be overridden on a per-policy basis using the **local-id** command. Refer to *local-id* on page 1640 for more information. |
| **policy** <*value*> | Creates an IKE policy, assigns the sequence number value of your choice, and enters the IKE Policy command set. Refer to section *IKE Policy Command Set* on page 1633 for more information. |

## Default Values

There are no default settings for this command.

## Command History

| | |
|---|---|
| Release 4.1 | Command was introduced. |

## Usage Examples

The following example creates an IKE policy with a policy priority setting of 1 and enters the IKE Policy for that policy:

(config)**#crypto ike policy 1**

## Technology Review

The following example configures an AOS product for VPN using IKE aggressive mode with pre-shared keys. The AOS product can be set to initiate IKE negotiation in main mode or aggressive mode. The product can be set to respond to IKE negotiation in main mode, aggressive mode, or any mode. In this example, the device is configured to initiate in aggressive mode and to respond to any mode.

This example assumes that the AOS product has been configured with a WAN IP address of 63.97.45.57 on interface **ppp 1** and a LAN IP address of 10.10.10.254 on interface **ethernet 0/1**. The peer private IP Subnet is 10.10.20.0.

For more detailed information on VPN configuration, refer to the technical support note *VPN Configuration Guide* located on the *ADTRAN OS System Documentation* CD provided with your unit.

Step 1:
Enter the Global configuration mode (i.e., config terminal mode).

>**enable**
**#configure terminal**

Step 2:
Enable VPN support using the **ip crypto** command. This command allows crypto maps to be applied to interfaces, and enables the IKE server to listen for IKE negotiation sessions on UDP port 500.

(config)**#ip crypto**

Step 3:
Set the local ID. During IKE negotiation, local IDs are exchanged between the local device and the peer device. In the AOS, the default setting for all local IDs are configured by the **crypto ike local-id** command. The default setting is for all local IDs to be the IPv4 address of the interface over which the IKE negotiation is occurring. In the future, a unique system-wide host name or fully qualified domain name could be used for all IKE negotiation.

(config)**#crypto ike local-id address**

Step 4:
Create IKE policy. In order to use IKE negotiation, an IKE policy must be created. Within the system, a list of IKE policies is maintained. Each IKE policy is given a priority number in the system. That priority number defines the position of that IKE policy within the system list. When IKE negotiation is needed, the system searches through the list, starting with the policy with priority of 1, looking for a match to the peer IP address.

An individual IKE policy can override the system local ID setting by having the **local-id** command specified in the IKE policy definition. This command in the IKE policy is used to specify the type of local ID and the local ID data. The type can be of IPv4 address, fully qualified domain name, or user-specified fully qualified domain name.

An IKE policy may specify one or more peer IP addresses that will be allowed to connect to this system. To specify multiple unique peer IP addresses, the **peer A.B.C.D** command is used multiple times within a single IKE policy. To specify that all possible peers can use a default IKE policy, the **peer any** command is given instead of the **peer A.B.C.D** command inside of the IKE policy. The policy with the **peer any** command specified will match to any peer IP address (and therefore should be given the highest numerical priority number). This will make the policy the last one to be compared against during IKE negotiation.

(config)**#crypto ike policy 10**
(config-ike)**#no local-id**
(config-ike)**#peer 63.105.15.129**
(config-ike)**#initiate aggressive**
(config-ike)**#respond anymode**
(config-ike)**#attribute 10**
(config-ike-attribute)**#encryption 3des**
(config-ike-attribute)**#hash sha**
(config-ike-attribute)**#authentication pre-share**
(config-ike-attribute)**#group 1**
(config-ike-attribute)**#lifetime 86400**

Step 5:
Define the remote ID settings. The **crypto ike remote-id** command is used to define the remote ID for a peer connecting to the system, specify the preshared-key associated with the specific remote ID, and (optionally) determine that the peer matching this remote ID should not use mode config (by using the **no-mode-config** keyword). Refer to *crypto ike remote-id* on page 471 for more information.

(config)**#crypto ike remote-id address 63.105.15.129 preshared-key mysecret123**

Step 6:
Define the transform-set. A transform set defines the encryption and/or authentication algorithms to be used to secure the data transmitted over the VPN tunnel. Multiple transform sets may be defined in a system. Once a transform set is defined, many different crypto maps within the system can reference it. In this example, a transform set named **highly_secure** has been created. This transform set defines ESP with authentication implemented using 3DES encryption and SHA1 authentication.

(config)**#crypto ipsec transform-set highly_secure esp-3des esp-sha-hmac**
(cfg-crypto-trans)**#mode tunnel**

Step 7:
Define an IP access list. An extended access control list is used to specify which traffic needs to be sent securely over the VPN tunnel. The entries in the list are defined with respect to the local system. The source IP address will be the source of the traffic to be encrypted. The destination IP address will be the receiver of the data on the other side of the VPN tunnel.

(config)**#ip access-list extended corporate_traffic**
(config-ext-nacl)**#permit ip 10.10.10.0 0.0.0.255 10.10.20.0 0.0.0.255 log deny ip any any**

Step 8:
Create crypto map. A crypto map is used to define a set of encryption schemes to be used for a given interface. A crypto map entry has a unique index within the crypto map set. The crypto map entry will specify whether IKE is used to generate encryption keys or if manually specified keys will be used. The crypto map entry will also specify who will be terminating the VPN tunnel, as well as which transform set or sets will be used to encrypt and/or authenticate the traffic on that VPN tunnel. It also specifies the lifetime of all created IPSec security associations.

(config)**#crypto map corporate_vpn 1 ipsec-ike**
(config-crypto-map)**#match address corporate_traffic**
(config-crypto-map)**#set peer 63.105.15.129**
(config-crypto-map)**#set transform-set highly_secure**
(config-crypto-map)**#set security-association lifetime kilobytes 8000**
(config-crypto-map)**#set security-association lifetime seconds 28800**
(config-crypto-map)**#no set pfs**

Step 9:
Configure a public interface. This process includes configuring the IP address for the interface and applying the appropriate crypto map to the interface. Crypto maps are applied to the interface on which encrypted traffic will be transmitted.

(config)**#interface ppp 1**
(config-ppp 1)**#ip address 63.97.45.57 255.255.255.248**
(config-ppp 1)**#crypto map corporate_vpn**
(config-ppp 1)**#no shutdown**

Step 10:
Configure a private interface. This process allows all traffic destined for the VPN tunnel to be routed to the appropriate gateway.

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#ip address 10.10.10.254 255.255.255.0**
(config-eth 0/1)**#no shutdown**
(config-eth 0/1)**#exit**

# crypto ike remote-id

Use the **crypto ike remote-id** command to specify the remote ID and to associate a pre-shared key with the remote ID. Use the **no** form of this command to disable these features.

*The AOS VPN feature must be enabled (using the **ip crypto** command) for the VPN tunnel to be activated.*

*For VPN configuration example scripts, refer to the technical support note **VPN Configuration Guide** located on the ADTRAN OS System Documentation CD provided with your unit.*

## Syntax Description

| | |
|---|---|
| **address** <*ip address*> | Specifies a valid remote IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| <*wildcard mask*> | Specifies the wildcard mask that corresponds to a range of IP addresses (network) or a specific host. Wildcard masks are expressed in dotted decimal notation (for example, 0.0.0.255). |
| **any** | Allows any remote ID (type and value). |
| **asn1-dn** <*name*> | Specifies an abstract syntax notation distinguished name as the remote ID (enter this value in LDAP format). |
| **crypto map** <*name*> <*number*> | Optional. Specifies the crypto map name and sequence number this remote ID corresponds to. |
| **fqdn** <*name*> | Specifies a fully qualified domain name (FQDN) (e.g., **adtran.com**) as the remote ID. |
| **ike policy** <*value*> | Optional. Specifies the IKE policy sequence number value this remote ID corresponds to. |
| **user-fqdn** <*name*> | Specifies a user fully qualified domain name or email address (e.g., **user1@adtran.com**) as the remote ID. |
| **preshared-key** <*key*> | Optional. Associates a preshared key with this remote ID. |
| **no-mode-config** | Optional. Specifies that the peer matching this remote ID should not use mode config. |
| **no-xauth** | Optional. Specifies that the peer matching this remote ID should not use Xauth. |
| **nat-t [v1 l v2] [allow l force l disable]** | Optional. Denotes whether peers matching this remote ID should allow, disable, or force NAT traversal versions 1 or 2. |

## Default Values

There are no default settings for this command.

## Command History

| | |
|---|---|
| Release 4.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include the **any**, **asn1-dn**, and **no-xauth** subcommands. |
| Release 7.1 | Command was expanded to include NAT traversal commands. |

## Functional Notes

The **fqdn** and **user-fqdn** <*fqdn*> line can include wildcard characters. The wildcard characters are "*****" for a 0 or more character match and "?" for a single character match. Currently, the "?" cannot be set up using the CLI, but it can be transferred to the unit via the startup-config.

Example for **user-fqdn**:

**john*@domain.com**

will match:

*johndoe@domain.com*
*johnjohn@adtran.comjohnjohn@myemail.com*
*john@adtran.comjohn@myemail.com*

Example for **fqdn**:

***.domain.com**

will match:

*www.domain.com*
*ftp.domain.com*
*one.www.domain.com*

The **address** remote ID can be in the form of a single host address or in the form of an IP address wildcard.

Example for **address** type:

**crypto ike remote id address 10.10.10.0 0.0.0.255**

will match:

*10.10.10.1*
*10.10.10.2*
*and all IP addresses in the form of 10.10.10.X (where X is 0 to 255)*

The **asn1-dn** <*name*> line can include wildcard characters. The wildcard characters are "*" for a 0 or more character match and "?" for a single character match. Currently, the "?" cannot be set up using the CLI, but it can be transferred to the unit via the startup-config.

Example for typical **asn1-dn** format with no wildcards:

**crypto ike remote-id asn1-dn "CN=MyRouter, C=US, S=ALCA, L=Huntsville, O=Adtran, OU=TechSupport"**
*(matches only remote ID strings with all fields exactly the same)*

Example for typical **asn1-dn** format with wildcards used to match a string within a field:

**crypto ike remote-id asn1-dn "CN=*, C=*, S=*, L=*, O=*, OU=*"**
*(matches any asn1-dn remote ID string from a peer)*

Example for typical **asn1-dn** format with wildcards used to match a portion of the remote ID:

**crypto ike remote-id asn1-dn "CN=*, C=US, S=ALCA, L=Huntsville, O=Adtran, OU=*"**
*(matches any remote ID string with the same values for the C, S, L, and O fields, and any values in the CN and OU fields)*

Example for typical **asn1-dn** format with wildcards used to match a portion of a field:

**crypto ike remote-id asn1-dn "CN=My*, C=US, S=ALCA, L=Huntsville, O=Adtran, OU=TechSupport"**
*(matches remote ID strings with all fields exactly the same, but with any CN field beginning with "My")*

## Usage Examples

The following example assigns a remote ID of 63.97.45.57 and associates the preshared key **mysecret** with the remote ID:

(config)#**crypto ike remote-id address 63.97.45.57 preshared-key mysecret**

# crypto ipsec transform-set *<name> <parameters>*

Use the **crypto ipsec transform-set** command to define the transform configuration for securing data (e.g., esp-3des, esp-sha-hmac, etc.). The transform set is then assigned to a crypto map using the map's **set transform-set** command. Refer to *set transform-set* on page 1611. Use the **no** form of this command to disable this feature.

> NOTE *For VPN configuration example scripts, refer to the technical support note **VPN Configuration Guide** located on the ADTRAN OS System Documentation CD provided with your unit.*

## Syntax Description

| | |
|---|---|
| *<name>* | Assigns a name to the transform set you are about to define. |
| *<parameters>* | Assigns a combination of up to three security algorithms from the following list:<br>• **ah-md5-hmac**, **ah-sha-hmac**<br>• **esp-des**, **esp-3des**, **esp-aes-128-cbc**, **esp-aes-192-cbc**, **esp-aes-256-cbc**, **esp-null**<br>• **esp-md5-hmac**, **esp-sha-hmac** |

## Default Values

There are no default settings for this command.

## Command History

| | |
|---|---|
| Release 4.1 | Command was introduced. |

## Functional Notes

Crypto map entries do not directly contain the transform configuration for securing data. Instead, the crypto map is associated with transform sets which contain specific security algorithms.

If no transform set is configured for a crypto map, the entry is incomplete and will have no effect on the system.

## Usage Examples

The following example first creates a transform set (**Set1**) consisting of two security algorithms (up to three may be defined), and then assigns the transform set to a crypto map (**Map1**):

(config)**#crypto ipsec transform-set Set1 esp-3des esp-sha-hmac**
(cfg-crypto-trans)**#exit**
(config)**#crypto map Map1 1 ipsec-ike**
(config-crypto-map)**#set transform-set Set1**

# crypto map

Use the **crypto map** command to define crypto map names and numbers and to enter the associated mode (either Crypto Map IKE or Crypto Map Manual). Use the **no** form of this command to disable this feature. Variations of this command include the following:

**crypto map** <*name*> <*index*> **ipsec-ike**
**crypto map** <*name*> <*index*> **ipsec-manual**

*For VPN configuration example scripts, refer to the technical support note **VPN Configuration Guide** located on the ADTRAN OS System Documentation CD provided with your unit.*

## Syntax Description

| | |
|---|---|
| <*name*> | Names the crypto map. You can assign the same name to multiple crypto maps, as long as the map index numbers are unique. |
| <*index*> | Assigns a crypto map sequence number. |
| **ipsec-ike** | Specifies the Crypto Map IKE (refer to *Crypto Map IKE Command Set* on page 1601). This supports IPSec entries that will use IKE to negotiate keys. |
| **ipsec-manual** | Specifies the Crypto Map Manual (refer to *Crypto Map Manual Command Set* on page 1612). This supports manually configured IPSec entries. |

## Default Values

There are no default settings for this command.

## Command History

| | |
|---|---|
| Release 4.1 | Command was introduced. |

## Functional Notes

Crypto map entries do not directly contain the transform configuration for securing data. Instead, the crypto map is associated with transform sets which contain specific security algorithms (refer to *crypto ipsec transform-set <name> <parameters>* on page 474).

Crypto map entries do not directly contain the selectors used to determine which data to secure. Instead, the crypto map entry refers to an access control list. An access control list is assigned to the crypto map using the **match address** command (refer to *ike-policy <number>* on page 1605).

If no transform set or access list is configured for a crypto map, the entry is incomplete and will have no effect on the system.

When you apply a crypto map to an interface (using the **crypto map** command within the interface's mode), you are applying all crypto maps with the given map name. This allows you to apply multiple crypto maps if you have created maps that share the same name but have different map index numbers.

## Usage Examples

The following example creates a new IPSec IKE crypto map called **testMap** with a map index of **10**:

(config)#**crypto map testMap 10 ipsec-ike**
(config-crypto-map)#

## Technology Review

A crypto map entry is a single policy that describes how certain traffic is to be secured. There are two types of crypto map entries: **ipsec-manual** and **ipsec-ike**. Each entry is given an index, which is used to sort the ordered list. When a nonsecured packet arrives on an interface, the crypto map set associated with that interface is processed in order. If a crypto map entry matches the nonsecured traffic, the traffic is discarded.

When a packet is to be transmitted on an interface, the crypto map set associated with that interface is processed in order. The first crypto map entry that matches the packet will be used to secure the packet. If a suitable security association (SA) exists, that is used for transmission. Otherwise, IKE is used to establish an SA with the peer. If no SA exists, and the crypto map entry is “respond only,” the packet is discarded.

When a secured packet arrives on an interface, its security parameter index (SPI) is used to look up an SA. If an SA does not exist, or if the packet fails any of the security checks (bad authentication, traffic does not match SA selectors, etc.), it is discarded. If all checks pass, the packet is forwarded normally.

# data-call

Use the **data-call** commands to set the pre-authentication defaults for inbound demand routing calls. For more detailed information on CHAP and PAP, refer to the *Technology Review* section of the command *ppp authentication* on page 1449. Use the **no** form of these commands to return to the default settings. Variations of this command include:

**data-call authentication protocol chap**
**data-call authentication protocol pap**
**data-call sent authentication protocol chap**
**data-call sent authentication protocol pap**

## Syntax Description

| | |
|---|---|
| **authentication protocol** | Sets the authentication protocol expected for inbound calls. |
| **sent authentication protocol** | Sets the authentication protocol sent for inbound calls. |
| **chap** | Configures CHAP authentication. |
| **pap** | Configures PAP authentication. |

## Default Values

By default, there is no configuration for authentication.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Functional Notes

There are certain PPP parameters that must be known before PPP can negotiate an inbound call when using demand routing. To ensure PPP convergence, it is recommended (in most cases) that demand routing interfaces use the same settings as those specified in the **data-call** commands. If the PPP parameters do not match the authenticated user, the link is renegotiated.

## Usage Examples

The following example sets the authentication protocol expected for incoming calls to CHAP. The router will then authenticate the peer using CHAP:

(config)**#data-call authentication protocol chap**

The following example sets the authentication protocol sent for incoming calls to PAP. This router may be authenticated by the peer using PAP:

(config)**#data-call sent authentication protocol pap**

# data-call

Use the **data-call mtu** command to set the pre-authentication defaults for maximum transmit unit (MTU) size. Use the **data-call multilink** command to enable multilink for inbound demand routing calls. Refer to the *mtu <size>* on page 1447 for more detailed syntax descriptions. Use the **no** form of each command to return to the default settings. Variations of this command include:

**data-call mtu** *<number>*
**data-call multilink**

## Syntax Description

| | |
|---|---|
| **mtu** *<number>* | Sets the maximum size for the transmit unit. Valid range is 64 to 1520. |
| **multilink** | Enables the negotiation of multilink MRU size for inbound calls. |

## Default Values

By default, the MTU size is 1500 and multilink is disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Functional Notes

There are certain PPP parameters that must be known before PPP can negotiate an inbound call when using demand routing. To ensure PPP convergence, it is recommended (in most cases) that demand routing interfaces use the same settings as those specified in the **data-call** commands. The **data-call mtu** *<number>* command sets the MTU and controls the negotiated maximum receive unit (MRU) size during incoming calls for link control protocol (LCP) negotiation. If the PPP parameters do not match the authenticated user, the link is renegotiated.

## Usage Examples

The following example specifies an MTU of 1200 on the demand routing interface:

(config)**#data-call mtu 1200**

The following example enables multilink for inbound demand routing calls:

(config)**#data-call multilink**

# enable password <*password*>

Use the **enable password** command to define a password (with optional encryption) for accessing the Enable mode. Use the **no enable password** command to remove a configured password. Variations of this command include:

**enable password md5** <*password*>
**enable password** <*password*>

*To prevent unauthorized users from accessing the configuration functions of your device, immediately define an Enable-level password.*

## Syntax Description

| | |
|---|---|
| **md5** | Optional. Specifies Message Digest 5 (MD5) as the encryption protocol to use when displaying the Enable password during show commands. If the **md5** keyword is not used, encryption is not used when displaying the Enable password during show commands. |
| <*password*> | Specifies the Enable password using a string (up to 30 characters in length). |

## Default Values

By default, there is no password configured for the Enable mode.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

To provide extra security, the AOS can encrypt the Enable password when displaying the current configuration. For example, the following is a **show configuration** printout (password portion) with an unencrypted Enable password (ADTRAN):

```
!
enable password ADTRAN
!
```

Alternately, the following is a **show configuration** printout (password portion) with an Enable password of ADTRAN using MD5 encryption:

```
!
enable password md5 encrypted 5aa5fbae7d01a90e79fb57705ce74676
!
```

# event-history on

Use the **event-history on** command to enable event logging for the AOS system. Event log messages will not be recorded unless this command has been issued (regardless of the **event-history priority** configured). The event log may be displayed using the **show event-history** command. Use the **no** form of this command to disable the event log.

## Syntax Description

No subcommands.

## Default Values

By default, the AOS event logging capabilities are disabled.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

The event history provides useful information regarding the status of the system and individual port states. Use the event history as a troubleshooting tool when identifying system issues. The following is a sample event history log.

**#show event-history**
Using 526 bytes
2002.07.12 15:34:01 T1.t1 1/1 Yellow
2002.07.12 15:34:01 INTERFACE_STATUS.t1 1/1 changed state to down.
2002.07.12 15:34:02 T1.t1 1/1 No Alarms
2002.07.12 15:34:02 INTERFACE_STATUS.t1 1/1 changed state to up.
2002.07.12 15:34:03 INTERFACE_STATUS.eth 0/1 changed state to up.
2002.07.12 15:34:10 OPERATING_SYSTEM Warm Start
2002.07.12 15:34:12 PPP.NEGOTIATION LCP up
2002.07.12 15:34:12 PPP.NEGOTIATION IPCP up

## Usage Examples

The following example enables the AOS event logging feature:

(config)**#event-history on**

# event-history priority

Use the **event-history priority** command to set the threshold for events stored in the event history. All events with the specified priority or higher will be kept for viewing in the local event log. The event log may be displayed using the **show event-history** command. Use the **no** form of this command to keep specified priorities from being logged. Variations of this command include:

**event-history priority error**
**event-history priority fatal**
**event-history priority info**
**event-history priority notice**
**event-history priority warning**

## Syntax Description

| | |
|---|---|
| **error** | Logs events with **error** and **fatal** priorities. |
| **fatal** | Logs only events with a **fatal** priority. |
| **info** | Logs all events. |
| **notice** | Logs events with **notice**, **warning**, **error**, and **fatal** priorities. |
| **warning** | Logs events with **warning**, **error**, and **fatal** priorities. |

## Default Values

By default, no event messages are logged to the event history.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

The event history provides useful information regarding the status of the system and individual port states. Use the event history as a troubleshooting tool when identifying system issues. The following is a sample event history log.

Router**#show event-history**
Using 526 bytes
2002.07.12 15:34:01 T1.t1 1/1 Yellow
2002.07.12 15:34:01 INTERFACE_STATUS.t1 1/1 changed state to down.
2002.07.12 15:34:02 T1.t1 1/1 No Alarms
2002.07.12 15:34:02 INTERFACE_STATUS.t1 1/1 changed state to up.
2002.07.12 15:34:03 INTERFACE_STATUS.eth 0/1 changed state to up.
2002.07.12 15:34:10 OPERATING_SYSTEM Warm Start
2002.07.12 15:34:12 PPP.NEGOTIATION LCP up
2002.07.12 15:34:12 PPP.NEGOTIATION IPCP up

## Usage Examples

The following example logs all events to the event history:

(config)**#event-history priority info**

# exception memory minimum <*value*>

Use the **exception memory minimum** command to initiate a reboot when the specified minimum amount of memory is no longer available. This ensures that adequate memory is available to store an exception report. Use the **no** form of this command to disable rebooting when the minimum memory limitation is violated.


*Executing the **exception memory minimum** command may cause the unit to reboot. ADTRAN recommends only using this command if advised to by ADTRAN Technical Support.*

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the minimum amount of memory (in bytes) that must be free before a reboot occurs. |

## Default Values

By default, **exception memory minimum** is disabled.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example sets the exception memory minimum to 3 Mbytes:

(config)**#exception memory minimum 30000000**

# exception report

Use the **exception report** command to specify the name of the output file for the exception report. Use the **no** form of this command to return to the default setting. Variations of this command include:

**exception report**
**exception report file-name** <*filename*>

## Syntax Description

| | |
|---|---|
| **file-name** <*filename*> | Optional. Specifies a file name for the exception report other than the default file name. |

## Default Values

By default, the exception report file name is **exception report-yyyyMMddHHmmss**. (The yyyyMMddHHmmss will be automatically replaced with the actual year, month, day, hour, minutes, and seconds.)

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example specifies **example** as the name of the output file for an exception report:

(config)**#exception report file-name example**
(config)**#exit**
**#exception report generate**
Exception report generated.
**#show flash**
1744 startup-config
45676 example-20050708080537
**#config t**
(config)**#no exception report file-name**
(config)**#exit**
Appropriate commands must be issued to preserve configuration.
**#exception report generate**
Exception report generated.
**#show flash**
1744 startup-config
45676 example-20050708080537
45900 exception-report-20050708080552

# ftp authentication *<name>*

Use the **ftp authentication** command to attach AAA login authentication lists to the File Transfer Protocol (FTP) server (refer to *aaa authorization commands <level>* on page 437 for more information). This list is only used if the authentication, authorization, and accounting (AAA) subsystem has been activated with the **aaa on** command. Use the **no** form of this command to return to the default setting.

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies the named list created with the **aaa authentication login** command. Enter **default** to use the AAA default login list. |

## Default Values

There is no default configuration for the list. If AAA is turned on but no **ftp authentication** list has been assigned, FTP denies all login attempts.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following example attaches the authentication list, **MyList**, to the FTP server:

(config)**#ftp authentication MyList**

The following example specifies that the AOS use the default AAA login list for FTP authentication:

(config)**#ftp authentication default**

# garp timer <*value*>

Use the **garp timer** command to adjust the timers used in all Generic Attribute Registration Protocol (GARP) applications (currently only GVRP) on the switch. Use the **no** form of this command to return to the default settings. Variations of this command include:

**garp timer join** <*value*>
**garp timer leave** <*value*>
**garp timer leaveall** <*value*>

## Syntax Description

| | |
|---|---|
| **join** <*value*> | Specifies the timer value (in milliseconds) between GARP application join messages. |
| **leave** <*value*> | Specifies the timer value (in milliseconds) between GARP application leave messages (must be at least three times longer than the join timer). |
| **leaveall** <*value*> | Specifies the timer value (in milliseconds) between GARP application leave all messages (must be greater than the leave timer). |

## Default Values

By default, the **join** timer is 200 milliseconds, the **leave** timer is 600 milliseconds, and the **leaveall** timer is 10,000 milliseconds.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Functional Notes

All devices communicating using GARP in the network need to have the same values for these timers. Changing these values is not recommended.

## Usage Examples

The following example specifies the time (in milliseconds) between GARP application **leave all** messages:

(config)**#garp timer leaveall 20000**

# gvrp

Use the **gvrp** command to enable GARP VLAN Registration Protocol (GVRP) on the switch globally. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, GVRP is disabled.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Functional Notes

Disabling GVRP globally will disable GVRP on all interfaces.

## Usage Examples

The following example enables GVRP on the switch globally:

(config)**#gvrp**

# hostname *<name>*

Use the **hostname** command to create a name used to identify the unit. This alphanumeric string should be used as a unique description for the unit. This string will be displayed in all prompts. Use the **no** form of this command to remove a host name.

## Syntax Description

| | |
|---|---|
| *<name>* | Identifies the unit using an alphanumeric string up to 32 characters. |

## Default Values

By default, the host name is **router**.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example creates a host name for the AOS device of **ATL_RTR** to identify the system as the Atlanta router:

(config)**#hostname ATL_RTR**

# interface *<interface>*

Use the **interface** command to activate the interface command set for the specified physical or virtual interface. Use the **no** form of this command to delete a configured interface. To activate the interface, enter the **no shutdown** command from within the specific interface command set. For example, (config-ppp 7)#**no shutdown**. Variations of this command include:

**interface** *<interface>*
**interface** *<interface>* **point-to-point**

## Syntax Description

| | |
|---|---|
| *<interface>* | Identifies the physical port type of the installed Network Interface Module (NIM), Dial-Backup Interface Module (DIM), or Ethernet port. Specify an interface in the format *<interface type [slot/port \| slot/port.sub-interface id \| interface id \| interface id.sub-interface id]>*. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **interface ?** for a complete list of valid interfaces. |
| **point-to-point** | Optional. Identifies the interface as a point-to-point link (versus multilink). Valid only on interfaces that support point-to-point (e.g., ATM and Frame Relay). By default, all created ATM and Frame Relay interfaces are point-to-point. |

## Default Values

No default values required for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 3.1 | Command expanded to include loopback interface. |
| Release 8.1 | Command expanded to include ATM interface. |
| Release 9.1 | Command expanded to include HDLC interface. |
| Release 11.1 | Command expanded to include demand, FXO, and PRI interfaces. |

## Usage Examples

The following example enters the serial interface mode for a serial module installed in slot 1:

(config)**#interface serial 1/1**
(config-ser 1/1)#

# interface range *<interface type> <slot/port>* - *<slot/port>*

Use the **interface range** command to enter configuration mode for a range of interfaces.

## Syntax Description

| | |
|---|---|
| *<interface type>* | Specifies the interface type (e.g., Ethernet, Gigabit Ethernet, etc.).Type **interface range ?** for a complete list of valid interfaces. |
| *<slot/port>* | Specifies the slot/port number of the first interface in the desired range of interfaces to be configured, followed by a hyphen (-) or a comma (,). |
| *<slot/port>* | Specifies the slot/port number of the last interface in the desired range of interfaces to be configured. |

## Default Values

No default value is necessary for this command.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |
| Release 11.1 | Command was expanded to include FXO range. |

## Functional Notes

All configuration changes made in this mode will apply to all interfaces in the range specified.

## Usage Examples

The following example selects seven consecutive E1 ports for configuration:

(config)**#interface range e1 3/1-3/7**
(config-eth 3/1-7)**#**

The following example selects nonconsecutive E1 ports for configuration:

(config)**#interface range e1 3/1-2, 3/4-6, 3/8**
(config-e1 3/1-2, 3/4-6, 3/8)**#**

# ip

Use the **ip** command to specify alternate transmission control protocol (TCP) ports for secure shell (SSH) and Telnet servers. Use the **no** form of this command to return to default settings. Variations of this command include:

**ip ssh-server** <*port*>
**ip telnet-server** <*port*>

## Syntax Description

| | |
|---|---|
| **ssh server** <*port*> | Configures the SSH server to listen on an alternate TCP port. |
| **telnet server** <*port*> | Configures the Telnet server to listen on an alternate TCP port. |

## Default Values

By default, the SSH server listens on TCP port 22 and Telnet listens on TCP port 23.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Functional Notes

SSH is a newer version of Telnet which allows you to run command line and graphical applications (as well as transfer files) over an encrypted connection.

## Usage Examples

The following example configures the Telnet server to listen on TCP port **2323** instead of the default port **23**:

(config)**#ip telnet-server 2323**

The following example configures the SSH server to listen on TCP port **2200** instead of the default port **22**:

(config)**#ip ssh-server 2200**

To return to the default settings, use the **no** version of the command. For example:

(config)**#no ip ssh-server 2200**

# ip access-list extended *<name>*

Use the **ip access-list extended** command to create an empty access list and enter the extended access-list command set. Use the **no** form of this command to delete an access list and all the entries contained in it. For more information on using access lists with the AOS firewall, refer to *ip policy-class <name>* on page 549. The following lists the complete syntax for the **ip access-list extended** commands:

**ip access-list extended** *<name>*
*<action> <protocol> <source> <source port> <destination> <destination port>*

## Syntax Description

| | | |
|---|---|---|
| *<name>* | Names the configured access list using an alphanumeric descriptor. All access list descriptors are case-sensitive. | |
| *<action>* | **permit** | Permits entry to the routing system for specified packets. |
| | **deny** | Denies entry to the routing system for specified packets. |
| | **remark** | Associates a descriptive tag (up to 80 alphanumeric characters enclosed in quotation marks) to the access list. Enter a functional description for the list such as “This list blocks all outbound Web traffic.” |
| *<protocol>* | Specifies the data protocol **ip**, **icmp**, **tcp**, **udp**, **ahp**, **esp**, **gre**, or a specific protocol. Range is 0 to 255. | |
| *<source>* | Specifies the source used for packet matching. Sources can be expressed in one of four ways:<br>1. Using the keyword **any** to match any IP address.<br>2. Using **host** *<ip address>* to specify a single host address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**).<br>3. Using the *<ip address> <wildcard mask>* format to match all IP addresses in a range. The wildcard mask that corresponds to a range of IP addresses (network) or a specific host. Wildcard masks are expressed in dotted decimal notation (for example, 0.0.0.255).<br>4. Using the keyword **hostname** to match based on a DNS name. The unit must be configured with DNS servers for this function to work. | |
| *<source port>* | Optional. The source port is used only when *<protocol>* is **tcp** or **udp**. The following keywords and port numbers/names are supported for the *<source port>* field: | |
| | **any** | Matches any destination port. |
| | **eq** *<port number/name>* | Matches only packets equal to specified port number. |
| | **gt** *<port number/name>* | Matches only packets with a port number greater than the specified port number. |
| | **lt** *<port number/name>* | Matches only packets with a port number less than the specified port number. |

| | |
|---|---|
| **neq** *<port number/name>* | Matches only packets that are not equal to the specified port number. |
| **range** *<port number/name>* | Matches only packets that contain a port number in the specified range. |
| *<port number>* | Specifies the port number used by TCP or UDP to pass information to upper layers using the following syntax: *<0-65535>*. All ports below 1024 are considered well-known ports and are controlled by the Internet Assigned Numbers Authority (IANA). All ports above 1024 are dynamically assigned ports that include registered ports for vendor-specific applications |
| *<port name>* | The following UDP port numbers can be specified using the associated names: |

| | |
|---|---|
| **biff** (Port 512) | **ntp** (Port 123) |
| **bootpc** (Port 68) | **pim-auto-rp** (Port 496) |
| **bootps**(Port 67) | **rip** (Port 520) |
| **discard** (Port 9) | **snmp** (Port 161) |
| **dnsix** (Port 195) | **snmptrap** (Port 162) |
| **domain** (Port 53) | **sunrpc** (Port 111) |
| **echo** (Port 7) | **syslog** (Port 514) |
| **isakmp** (Port 500) | **tacacs** (Port 49) |
| **mobile-ip** (Port 434) | **talk** (Port 517) |
| **nameserver** (Port 42) | **tftp** (Port 69) |
| **netbios-dgm** (Port 138) | **time** (Port 37) |
| **netbios-ns** (Port 137) | **who** (Port 513) |
| **netbios-ss** (Port 139) | **xdmcp** (Port 177) |

The following TCP port numbers can be specified using the associated names:

| | |
|---|---|
| **bgp** (Port 179) | **lpd** (Port 515) |
| **chargen** (Port 19) | **nntp** (Port 119) |
| **cmd** (Port 514) | **pim-auto-rp** (Port 496) |
| **daytime** (Port 13) | **pop2** (Port 109) |
| **discard** (Port 9) | **pop3** (Port 110) |
| **domain** (Port 53) | **smtp** (Port 25) |
| **echo** (Port 7) | **sunrpc** (Port 111) |
| **exec** (Port 512) | **syslog** (Port 514) |
| **finger** (Port 79) | **tacacs** (Port 49) |
| **ftp** (Port 21) | **talk** (Port 517) |
| **gopher** (Port 70) | **tftp** (Port 69) |

| | |
|---|---|
| **hostname** (Port 101) | **telnet** (Port 23) |
| **ident** (Port 113) | **time** (Port 37) |
| **irc** (Port 194) | **uucp** (Port 540) |
| **klogin** (Port 543) | **whois** (Port 43) |
| **kshell** (Port 544) | **www** (Port 80) |
| **login** (Port 513) | |

*<destination>* Specifies the destination used for packet matching. Destinations can be expressed in one of four ways:

1. Using the keyword **any** to match any IP address.
2. Using **host** *<ip address>* to specify a single host address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**).
3. Using the *<ip address>* *<wildcard mask>* format to match all IP addresses in a range. The wildcard mask that corresponds to a range of IP addresses (network) or a specific host. Wildcard masks are expressed in dotted decimal notation (for example, 0.0.0.255).
4. Using the keyword **hostname** to match based on a DNS name. The unit must be configured with DNS servers for this function to work.

*<destination port>* Optional. Specifies the destination port. Only valid when *<protocol>* is **tcp** or **udp.** The same keywords and port numbers/names used for the *<source port>* field are valid for the *<destination port>* field. Refer to previously listed *<source port>* for more details.

## Default Values

By default, all AOS security features are disabled and there are no configured access lists.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |

## Functional Notes

Access control lists (ACLs) are used as packet selectors by different AOS features (firewall, VPN, QoS); by themselves they do nothing. ACLs are composed of an ordered list of entries with an implicit **deny all** at the end of each list. An ACL entry contains two parts: an action (**permit** or **deny**) and a packet pattern. A **permit** ACL is used to match packets (meeting the specified pattern) to enter the router system. A **deny** ACL advances the AOS to the next access policy entry. AOS provides two types of ACLs: standard and extended. Standard ACLs match based on the source of the packet. Extended ACLs match based on the source and destination of the packet.

ACLs are performed in order from the top of the list down. Generally, the most specific entries should be at the top and the more general at the bottom.

## Usage Examples

The following example creates an access list **AllowIKE** to allow all IKE (UDP Port 500) packets from the 190.72.22.55.0/24 network:

(config)**#ip access-list extended AllowIKE**
(config-ext-nacl)**#permit udp 190.72.22.55.0 0.0.0.255 eq 500 any eq 500**

For more details, refer to the *ADTRAN OS System Documentation* CD or the ADTRAN website (*www.adtran.com*) for technical support notes regarding access-list configuration.

# ip access-list standard <*name*>

Use the **ip access-list standard** command to create an empty access list and enter the standard access-list command set. Use the **no** form of this command to delete an access list and all the entries contained in it. For more information on using access lists with the AOS firewall, refer to *ip policy-class* <*name*> on page 549. The following lists the complete syntax for the **ip access-list standard** commands:

**ip access-list standard** <*listname*>
<*action*> <*source*>

## Syntax Description

| | | |
|---|---|---|
| <*name*> | Identifies the configured access list using an alphanumeric descriptor. All access list descriptors are case-sensitive. | |
| <*action*> | **permit** | Permits entry to the routing system for specified packets. |
| | **deny** | Denies entry to the routing system for specified packets. |
| | **remark** | Associates a descriptive tag (up to 80 alphanumeric characters enclosed in quotation marks) to the access list. Enter a functional description for the list such as “This list blocks all outbound Web traffic.” |
| <*source*> | Specifies the source used for packet matching. Sources can be expressed in one of four ways:<br>1. Using the keyword **any** to match any IP address.<br>2. Using **host** <*ip address*> to specify a single host address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**).<br>3. Using the <*ip address*> <*wildcard mask*> format to match all IP addresses in a range. The wildcard mask that corresponds to a range of IP addresses (network) or a specific host. Wildcard masks are expressed in dotted decimal notation (for example, 0.0.0.255).4.Using the keyword **hostname** to match based on a DNS name. The unit must be configured with DNS servers for this function to work. | |

## Default Values

By default, all AOS security features are disabled and there are no configured access lists.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |

## Functional Notes

Access control lists (ACLs) are used as packet selectors by different AOS features (firewall, VPN, QoS); by themselves they do nothing. ACLs are composed of an ordered list of entries with an implicit **deny all** at the end of each list. An ACL entry contains two parts: an action (**permit** or **deny**) and a packet pattern. A **permit** ACL is used to match packets (meeting the specified pattern) to enter the router system. A **deny** ACL advances the AOS to the next access policy entry. AOS provides two types of ACLs: standard and extended. Standard ACLs match based on the source of the packet. Extended ACLs match based on the source and destination of the packet.

ACLs are performed in order from the top of the list down. Generally, the most specific entries should be at the top and the more general at the bottom.

## Usage Examples

The following example creates an access list **UnTrusted** to deny all packets from the 190.72.22.248/30 network:

(config)**#ip access-list standard UnTrusted**
(config-std-nacl)**#deny 190.72.22.248 0.0.0.3**

For more details, refer to the *ADTRAN OS System Documentation* CD or the ADTRAN website (www.adtran.com) for technical support notes regarding access list configuration.

# ip as-path-list *<name>*

Use the **ip as-path-list** command to create IP autonomous system (AS) path lists for route-map use. Use the **no** form of this command to delete the AS path list.

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies the name of the AS path list. Refer to *AS Path List Configuration Command Set* on page 1646 for more information of the available options. |

## Default Values

By default, no AS path lists are defined.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Functional Notes

AS path lists are a type of route filter that permits or denies BGP routes based on the AS_PATH attribute. AS path lists define a list of AS specifications that once created, may then be referenced in a route map. Refer to the *Usage Examples* section below.

## Usage Examples

The following example creates the AS path list **list5** and enters the IP **as-path-list** command mode:

(config)**#ip as-path-list list5**
(config-as-path-list)#

# ip classless

Use the **ip classless** command to forward classless packets to the best supernet route available. A classless packet is a packet addressed for delivery to a subnet of a network with no default network route.

## Syntax Description

No subcommands.

## Default Values

By default, this command is enabled.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

AOS products only function in classless mode. You cannot disable this feature.

## Usage Examples

The following example enables the system to forward classless packets:

(config)**#ip classless**

# ip community-list <*name*>

Use the **ip community-list** command to create a community list for BGP route map use. Use the **no** form of this command to delete a community list.

## Syntax Description

| | |
|---|---|
| <*name*> | Specifies the name of the community to use in the community list attribute for BGP routes. Refer to *Community List Configuration Command Set* on page 1679 for more information of the available options. |

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example creates a community list **MyList** and enters the Community List Configuration mode:

(config)**#ip community-list MyList**

# ip crypto

Use the **ip crypto** command to enable AOS VPN functionality and allow crypto maps to be added to interfaces. Use the **no** form of this command to disable the VPN functionality.Variations of this command include:

**ip crypto**
**ip crypto fast-failover**

*Disabling the AOS security features (using the **no ip crypto** command) does not affect VPN configuration settings (with the exception of the removal of all crypto maps from the interfaces). All other configuration parameters will remain intact, and VPN functionality will be disabled.*

*For VPN configuration example scripts, refer to the **VPN Configuration Guide** located on the ADTRAN OS System Documentation CD provided with your unit.*

## Syntax Description

| | |
|---|---|
| **fast-failover** | Optional. This setting is used when the same crypto map is applied to two different egress interfaces. It allows the quick deletion of IKE and IPSec SAs when the default route policy-class changes. |

## Default Values

By default, all AOS VPN functionality is disabled.

## Command History

| | |
|---|---|
| Release 4.1 | Command was introduced. |
| Release 11.2 | Command was expanded to include the fast-failover feature. |

## Functional Notes

VPN-related settings will not go into effect until you enable VPN functionality using the **ip crypto** command. The AOS allows you to perform all VPN-related configuration prior to enabling **ip crypto**, with the exception of assigning a **crypto map** to an interface. The **no ip crypto** command removes all crypto maps from the interfaces. Enabling **ip crypto** enables the IKE server on UDP Port 500. The **no** form of this command disables the IKE server on UDP Port 500.

## Usage Examples

The following example enables VPN functionality:

(config)**#ip crypto**

# ip default-gateway *<ip address>*

Use the **ip default-gateway** command to specify a default gateway on a switch or on a router if (and only if) IP routing is NOT enabled on the router. Use the **ip route** command to add a default route to the route table when using IP routing functionality. Refer to *ip route* on page 564 for more information. Use the **no** form of this command to disable this feature.

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies the default gateway IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

## Default Values

By default, there is no configured default gateway.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example disables IP routing on a router and configures a default gateway for **10.10.10.1**:

(config)**#no ip routing**
(config)**#ip default-gateway 10.10.10.1**

The following example specifies a default gateway for the management interface on a switch:

(config)**#ip default-gateway 10.10.10.1**

# ip dhcp-server database local

Use the **ip dhcp-server database local** command to configure a Dynamic Host Configuration Protocol (DHCP) database agent with local bindings. Use the **no** form of this command to disable this option.

## Syntax Description

No subcommands.

## Default Values

No default values.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example configures the DHCP database agent with local bindings:

(config)**#ip dhcp-server database local**

# ip dhcp-server excluded-address *<start ip address> <end ip address>*

Use the **ip dhcp-server excluded-address** command to specify IP addresses that cannot be assigned to Dynamic Host Configuration Protocol (DHCP) clients. Use the **no** form of this command to remove a configured IP address restriction.

## Syntax Description

| | |
|---|---|
| *<start ip address>* | Specifies the lowest IP address in the range OR a single IP address to be excluded. |
| *<end ip address>* | Optional. Specifies the highest IP address in the range. This field is not required when specifying a single IP address.<br>IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

## Default Values

By default, there are no excluded IP addresses.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |

## Functional Notes

The AOS DHCP server (by default) allows all IP addresses for the DHCP pool to be assigned to requesting clients. This command is used to ensure that the specified address is never assigned by the DHCP server. When static addressed hosts are present in the network, it is helpful to exclude the IP addresses of the host from the DHCP IP address pool. This will avoid IP address overlap.

## Usage Examples

The following example excludes an IP address of 172.22.5.100 and the range 172.22.5.200 through 172.22.5.250:

(config)**#ip dhcp-server excluded-address 172.22.5.100**
(config)**#ip dhcp-server excluded-address 172.22.5.200 172.22.5.250**

# ip dhcp-server ping packets *<number>*

Use the **ip dhcp-server ping packets** command to specify the number of ping packets the Dynamic Host Configuration Protocol (DHCP) server will transmit before assigning an IP address to a requesting DHCP client. Transmitting ping packets verifies that no other hosts on the network are currently configured with the specified IP address. Use the **no** form of this command to prevent the DHCP server from using ping packets as part of the IP address assignment process.

## Syntax Description

| | |
|---|---|
| *<number>* | Specifies the number of DHCP ping packets sent on the network before assigning the IP address to a requesting DHCP client. |

## Default Values

By default, the number of DHCP server ping packets is set at 2 packets.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |

## Functional Notes

Before assigning an IP address to a requesting client, the AOS DHCP server transmits a ping packet on the network to verify there are no other network hosts already configured with the specified address. If the DHCP server receives no reply, the IP address is assigned to the requesting client and added to the DHCP database as an assigned address. Configuring the **ip dhcp-server ping packets** command with a value of **0** prevents the DHCP server from using ping packets as part of the IP address assignment process.

## Usage Examples

The following example configures the DHCP server to transmit four ping packets before assigning an address:

(config)**#ip dhcp-server ping packets 4**

# ip dhcp-server ping timeout <*value*>

Use the **ip dhcp-server ping timeout** command to specify the interval (in milliseconds) the Dynamic Host Configuration Protocol (DHCP) server will wait for a response to a transmitted DHCP ping packet. The DHCP server transmits ping packets before assigning an IP address to a requesting DHCP client. Transmitting ping packets verifies that no other hosts on the network are currently configured with the specified IP address. Use the **no** form of this command to return to the default timeout interval.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the number of milliseconds the DHCP server will wait for a response to a transmitted DHCP ping packet. Valid range is 1 to 1000 milliseconds. |

## Default Values

By default, the **ip dhcp-server ping timeout** is set to 500 milliseconds.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |

## Functional Notes

Before assigning an IP address to a requesting client, the AOS DHCP server transmits a ping packet on the network to verify there are no other network hosts already configured with the specified address. If the DHCP server receives no reply, the IP address is assigned to the requesting client and added to the DHCP database as an assigned address.

## Usage Examples

The following example configures the DHCP server to wait 900 milliseconds for a response to a transmitted DHCP ping packet before considering the ping a failure:

(config)**#ip dhcp-server ping timeout 900**

# ip dhcp-server pool *<name>*

Use the **ip dhcp-server pool** command to create a Dynamic Host Configuration Protocol (DHCP) address pool and enter the DHCP pool. Use the **no** form of this command to remove a configured DHCP address pool. Refer to the section *DHCP Pool Command Set* on page 1966 for more information.

## Syntax Description

| | |
|---|---|
| *<name>* | Identifies the configured DHCP server address pool using an alphanumeric string (up to 32 characters in length). |

## Default Values

By default, there are no configured DHCP address pools.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |

## Functional Notes

Use the **ip dhcp-server pool** to create multiple DHCP server address pools for various segments of the network. Multiple address pools can be created to service different segments of the network with tailored configurations.

## Usage Examples

The following example creates a DHCP server address pool (labeled **SALES**) and enters the DHCP server pool mode:

```
(config)#ip dhcp-server pool SALES
(config-dhcp)#
```

# ip domain-lookup

Use the **ip domain-lookup** command to enable the IP domain naming system (DNS), allowing DNS-based host translation (name-to-address). Use the **no** form of this command to disable DNS.

## Syntax Description

No subcommands.

## Default Values

By default, this command is enabled.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Functional Notes

Use the **ip domain-lookup** command to enable the DNS client in the router. This will allow the user to input Web addresses instead of IP addresses for applications such as ping, Telnet, and traceroute.

## Usage Examples

The following example enables DNS:

(config)**#ip domain-lookup**

# ip domain-name *<name>*

Use the **ip domain-name** command to define a default IP domain name to be used by the AOS to resolve host names. Use the **no** form of this command to disable this function.

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies the default IP domain name used to resolve unqualified host names. Do not include the initial period that separates the unresolved name from the default domain name. |

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Functional Notes

Use the **ip domain-name** command to set a default name which will be used to complete any IP host name that is invalid (i.e., any name that is not recognized by the name server). When this command is enabled, any IP host name that is not initially recognized will have the **ip domain-name** appended to it and the request will be resent.

## Usage Examples

The following example defines **adtran** as the default domain name:

(config)**#ip domain-name adtran**

# ip domain-proxy

Use the **ip domain-proxy** command to enable DNS proxy for the router. This enables the router to act as a proxy for other units on the network. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Functional Notes

When this command is enabled, incoming DNS requests will be handled by the router. It will first search its host table for the query, and if it is not found there the request will be forwarded to the servers configured with the **ip name-server** command.

## Usage Examples

The following example enables DNS proxy:

(config)**#ip domain-proxy**

# ip ffe max-entries *<entries>*

Use the **ip ffe max-entries** command to set the global maximum number of FastFlow Engine (FFE) entries allowed. Use the **no** form of these command to return to the default value.

*Issuing this command will cause all FastFlow entries in the unit to be cleared.*

## Syntax Description

| | |
|---|---|
| *<entries>* | Specifies the total number of FFE entries for all interfaces. Valid range is 1 to 32,768. |

## Default Values

By default, the **ip ffe max-entries** is set to 16,384.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example sets the total maximum number of FFE entries to **500**:

(config)**#ip ffe max-entries 500**

# ip ffe timeout

Use the **ip ffe timeout** command to set the time-to-live for FastFlow Engine (FFE) entries based on their IP protocol. Use the **no** form of these commands to return to the default value. Variations of this command include:

**ip ffe timeout ah** <*max timeout*>
**ip ffe timeout ah** <*max timeout*> <*inactive timeout*>
**ip ffe timeout esp** <*max timeout*>
**ip ffe timeout esp** <*max timeout*> <*inactive timeout*>
**ip ffe timeout icmp** <*max timeout*>
**p ffe timeout icmp** <*max timeout*> <*inactive timeout*>
**ip ffe timeout other** <*max timeout*>
**ip ffe timeout other** <*max timeout*> <*inactive timeout*>
**ip ffe timeout tcp** <*max timeout*>
**ip ffe timeout tcp** <*max timeout*> <*inactive timeout*>
**ip ffe timeout udp** <*max timeout*>
**ip ffe timeout udp** <*max timeout*> <*inactive timeout*>

## Syntax Description

| | |
|---|---|
| **ah** | Specifies timeout values in seconds for authentication header protocol (AH). |
| **esp** | Specifies timeout values in seconds for encapsulated security protocol (ESP). |
| **icmp** | Specifies timeout values in seconds for internet control message protocol (ICMP). |
| **other** | Specifies timeout values in seconds for all protocols not listed. |
| **tcp** | Specifies timeout values in seconds for transmission control protocol (TCP). |
| **udp** | Specifies timeout values in seconds for user datagram protocol (UDP). |
| <*max timeout*> | Specifies maximum age timeout in seconds. This is the maximum amount of time an entry will be kept in the FFE table regardless of activity. Valid range is 60 to 86,400 seconds. |
| <*inactive timeout*> | Optional. Specifies idle timeout in seconds. This is the amount of time an entry will remain in the FFE table with no additional activity. Valid range is 10 to 86,400 seconds. |

## Default Values

By default, the maximum age timeouts are set to 1800 seconds and the inactive timeouts are set to 15 seconds.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example sets time to live (TTL) for TCP packets to **1000** seconds.

(config)**# ip ffe timeout tcp 1000**

# ip firewall

Use the **ip firewall** command to enable AOS security features including access control policies and lists, Network Address Translation (NAT), and the stateful inspection firewall. Use the **no** form of this command to disable the security functionality.

*Disabling the AOS security features (using the **no ip firewall** command) does not affect security configuration. All configuration parameters will remain intact, but no security data processing will be attempted.*

*For information regarding the use of OSPF with **ip firewall** enabled, refer to the **Functional Note** for router ospf* *on page 652*.

*Regarding the use of IKE negotiation for VPN with **ip firewall** enabled, there can be up to six channel groups with 2 to 8 interfaces per group. Dynamic protocols are not yet supported (only static). A physical interface can be a member of only one channel group.*

## Syntax Description

No subcommands.

## Default Values

By default, all AOS security features are disabled.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |

## Functional Notes

This command enables firewall processing for all interfaces with a configured policy class. Firewall processing consists of the following functions:

**Attack Protection:** Detects and discards traffic that matches profiles of known networking exploits or attacks.

**Session Initiation Control:** Allows only sessions that match traffic patterns permitted by access-control policies to be initiated through the router.

**Ongoing Session Monitoring and Processing:** Each session that has been allowed through the router is monitored for any irregularities that match patterns of known attacks or exploits. This traffic will be dropped. Also, if NAT is configured, the firewall modifies all traffic associated with the session according to the translation rules defined in NAT access policies. Finally, if sessions are inactive for a user-specified amount of time, the session will be closed by the firewall.

**Application Specific Processing:** Certain applications need special handling to work correctly in the presence of a firewall. AOS uses application-level gateways (ALGs) for these applications.

The AOS includes several security features to provide controlled access to your network. The following features are available when security is enabled (using the **ip firewall** command):

1. Stateful Inspection Firewall

The AOS (and your unit) act as an ALG and employ a stateful inspection firewall that protects an organization's network from common cyber attacks including TCP syn-flooding, IP spoofing, ICMP redirect, land attacks, ping-of-death, and IP reassembly problems. In addition, further security is added with use of Network Address Translation (NAT) and Port Address Translation (PAT) capability.

2. Access Policies

AOS access control policies (ACPs) are used to allow, discard, or manipulate (using **NAT**) data for each physical interface. Each ACP consists of a selector (**access list**) and an action (**allow**, **discard**, **NAT**). When packets are received on an interface, the configured ACPs are applied to determine whether the data will be processed or discarded.

3. Access Lists

Access control lists (ACLs) are used as packet selectors by ACPs; by themselves they do nothing. ACLs are composed of an ordered list of entries. Each entry contains two parts: an action (**permit** or **deny**) and a packet pattern. A permit ACL is used to permit packets (meeting the specified pattern) to enter the router system. A deny ACL advances the AOS to the next access policy entry. The AOS provides two types of ACLs: **standard** and **extended**. Standard ACLs allow source IP address packet patterns only. Extended ACLs may specify patterns using most fields in the IP header and the TCP or UDP header.

## Usage Examples

The following example enables the AOS security features:

(config)**#ip firewall**

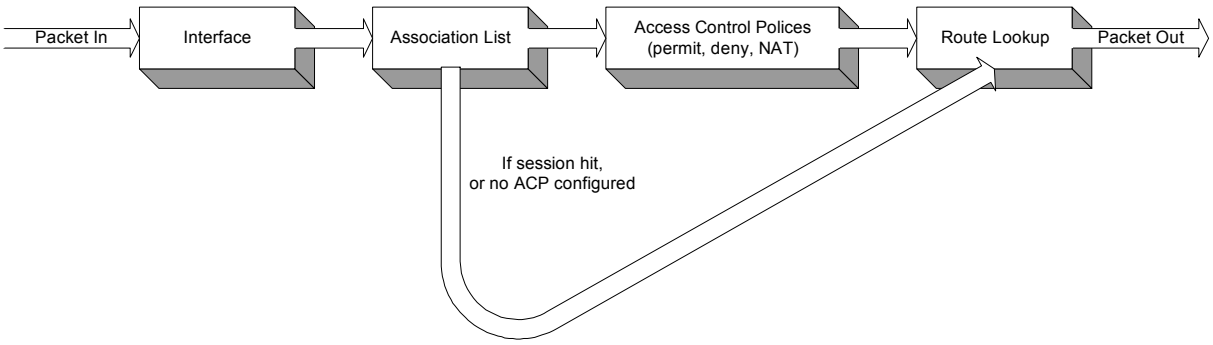
## Technology Review

**Concepts:** Access control using the AOS firewall has two fundamental parts: Access Control Lists (ACLs) and Access Policy Classes (ACPs). ACLs are used as packet selectors by other AOS systems; by themselves they do nothing. ACPs consist of a selector (ACL) and an action (allow, discard, NAT). ACPs integrate both allow and discard policies with NAT. ACPs have no effect until they are assigned to a network interface.

Both ACLs and ACPs are order dependent. When a packet is evaluated, the matching engine begins with the first entry in the list and progresses through the entries until it finds a match. The first entry that matches is executed.

**Packet Flow:**




**Case 1: Packets from interfaces with a configured policy class to any other interface**

ACPs are applied when packets are received on an interface. If an interface has not been assigned a policy class, by default it will allow all received traffic to pass through. If an interface has been assigned a policy class but the firewall has not been enabled with the **ip firewall** command, traffic will flow normally from this interface with no firewall processing.

**Case 2: Packets that travel in and out a single interface with a configured policy class**

These packets are processed through the ACPs as if they are destined for another interface (identical to Case 1).

**Case 3: Packets from interfaces without a configured policy class to interfaces with one**

These packets are routed normally and are not processed by the firewall. The **ip firewall** command has no effect on this traffic.

**Case 4: Packets from interfaces without a configured policy class to other interfaces without a configured policy class**

This traffic is routed normally. The **ip firewall** command has no effect on this traffic.

Attack Protection:

When the **ip firewall** command is enabled, firewall attack protection is enabled. The AOS blocks traffic (matching patterns of known networking exploits) from traveling through the device. For some of these attacks, the user may manually disable checking/blocking while other attack checks are always on anytime the firewall is enabled.

The table (on the following pages) outlines the types of traffic discarded by the firewall attack protection engine. Many attacks use similar invalid traffic patterns; therefore attacks other than the examples listed below may also be blocked by the firewall. To determine if a specific attack is blocked by the AOS firewall, please contact ADTRAN technical support.

| Invalid Traffic Pattern | Manually Enabled? | AOS Firewall Response | Common Attacks |
|---|---|---|---|
| Larger than allowed packets | No | Any packets that are longer than those defined by standards will be dropped. | Ping of Death |
| Fragmented IP packets that produce errors when attempting to reassemble | No | The firewall intercepts all fragments for an IP packet and attempts to reassemble them before forwarding to destination. If any problems or errors are found during reassembly, the fragments are dropped. | SynDrop, TearDrop, OpenTear, Nestea, Targa, Newtear, Bonk, Boink |
| Smurf Attack | No | The firewall will drop any ping responses that are not part of an active session. | Smurf Attack |
| IP Spoofing | No | The firewall will drop any packets with a source IP address that appears to be spoofed. The IP route table is used to determine if a path to the source address is known (out of the interface from which the packet was received). For example, if a packet with a source IP address of 10.10.10.1 is received on interface fr 1.16 and no route to 10.10.10.1 (through interface fr 1.16) exists in the route table, the packet is dropped. | IP Spoofing |
| ICMP Control Message Floods and Attacks | No | The following types of ICMP packets are allowed through the firewall: echo, echo-reply, TTL expired, dest. Unreachable, and quench. These ICMP messages are only allowed if they appear to be in response to a valid session. All others are discarded. | Twinge |
| Attacks that send TCP URG packets | Yes | Any TCP packets that have the URG flag set are discarded by the firewall. | Winnuke, TCP XMAS Scan |
| Falsified IP Header Attacks | No | The firewall verifies that the packet's actual length matches the length indicated in the IP header. If it does not, the packet is dropped. | Jolt/Jolt2 |
| Echo | No | All UDP echo packets are discarded by the firewall. | Char Gen |

| Invalid Traffic Pattern | Manually Enabled? | AOS Firewall Response | Common Attacks |
|---|---|---|---|
| Land Attack | No | Any packets with the same source and destination IP addresses are discarded. | Land Attack |
| Broadcast Source IP | No | Packets with a broadcast source IP address are discarded. | |
| Invalid TCP Initiation Requests | No | TCP SYN packets that have ack, urg rst, or fin flags set are discarded. | |
| Invalid TCP Segment Number | No | The sequence numbers for every active TCP session are maintained in the firewall session database. If the firewall received a segment with an unexpected (or invalid) sequence number, the packet is dropped. | |
| IP Source Route Option | No | All IP packets containing the IP source route option are dropped. | |

## Application Specific Processing

The following applications and protocols require special processing to operate concurrently with NAT/firewall functionality. The AOS firewall includes ALGs for handling these applications and protocols:

AOL Instant Messenger (AIM®)
VPN ALGS: ESP and IKE
FTP
H.323: H.245 Q.931 ASN1 PER decoding and Encoding
ICQ®
IRC
Microsoft® Games
Net2Phone
PPTP
Quake®
Real-Time Streaming Protocol
SMTP
HTTP
CUseeme
SIP
L2TP
PcAnywhere™
SQL
Microsoft Gaming Zone

To determine if a specific application requires special processing, contact technical support at *www.adtran.com.*

# ip firewall alg

Use the **ip firewall alg** command to enable the application-level gateway (ALG) for a particular application. Use the **no** form of this command to disable ALG for the application. Variations of this command include the following:

**ip firewall alg ftp**
**ip firewall alg h323**
**ip firewall alg pptp**
**ip firewall alg sip**
**ip firewall alg sip udp** <*number*>

## Syntax Description

| | |
|---|---|
| **ftp** | Enables the FTP ALG. |
| **h323** | Enables the H.323 ALG. H.323 is a generic recommendation from the ITU that sets standards for multimedia communications over networks without guaranteed Quality of Service (QoS). |
| **pptp** | Enables the PPTP ALG. |
| **sip** | Enables the SIP ALG. |
| **udp** <*number*> | Optional. Specifies a UDP port for the SIP ALG. Valid range is 1 to 65,535. Multiple UDP ports can be entered. |

## Default Values

By default, the ALG for FTP, H323, PPTP, and SIP are enabled and the UDP port is set to 5060.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |
| Release 10.1 | Command was expanded to include H323. |
| Release 11.1 | Command was expanded to include SIP UDP <*port#*>. |

## Functional Notes

Enabling the application layer gateway (ALG) for a specific protocol gives the firewall additional information about that complex protocol and causes the firewall to perform additional processing for packets of that protocol. When the ALG is disabled, the firewall treats the complex protocol as any other simple protocol. The firewall needs no special knowledge to work well with simple protocols.


*Disabling the IP firewall ALG may cause the firewall to block some of the traffic for the specified protocol.*

**Session Initiation Protocol (SIP) ALG Information**

By default, the AOS SIP ALG is enabled. This ALG allows the firewall to examine the ALL SIP packets it identifies and maintain knowledge of SIP transmissions on the network based on the SIP header. The SIP ALG requires the use of the SIP stack and the SIP proxy server in order to properly route SIP calls and maintain the SIP information. When the SIP ALG is enabled, the SIP stack and SIP proxy server are automatically enabled. For proper SIP operation, the firewall must also be configured to allow for dynamic holes for the RTP/RTCP traffic associated with SIP calls between User Agents (UAs). This functionality must be manually enabled using the **ip rtp firewall-traversal** command.

To completely disable SIP operation in AOS, the following commands should be entered: **no ip firewall alg sip**, **no ip sip**, and **no ip rtp firewall-traversal**. The **no ip firewall alg sip** command disables the SIP ALG. The **no ip sip** command disables the SIP stack and frees all memory allocated to the stack.

## Usage Examples

The following example disables ALG for FTP:

(config)**#no ip firewall alg ftp**

The following example enables port **1020** for UDP:

(config)**#ip firewall alg sip udp 1020**

## Technology Review

SIP is one protocol in a suite of protocols that was designed to replace H.323 for IP telephony. SIP operates in Layer 7 of the OSI model (application layer) to create, modify, and terminate sessions between nodes. SIP not only provides recommendations for IP telephony, but multimedia distribution and conferences as well. SIP version 1.0 was defined in RFC2453, and was refined to SIP version 2.0 in RFC3261.

SIP operations occur between SIP UAs and SIP servers. Types of SIP servers include proxy, redirect, registrar, and presence. The part of a SIP UA that sends messages is known as the User Agent Client (UAC). The part of a SIP UA that receives messages is known as a User Agent Server (UAS).

SIP was originally designed for use over User Datagram Protocol (UDP). SIP servers, by default, listen on port 5060. Due to security concerns, SIP is now transitioning to TCP and Transport Layer Security (TLS). SIP servers using TLS-over-TCP listen on port 5061. SIP UAs listen on a range of ports. The listening UDP port can be manually changed using the **ip firewall alg sip udp** command.

SIP uses the Session Description Protocol (SDP) to format the SIP message body in order to negotiate a Real-time Transport Protocol (RTP)/Real-time Transport Control Protocol (RTCP) connection between two or more UAs. The ports used for this will always be selected in a pair, with the even port used for RTP and the odd port for RTCP. SIP, because it uses SDP and RTP, causes many problems for standard firewalls. Neither SIP nor RTP are guaranteed to be symmetric, thus causing problems for stateful inspection firewalls that rely on symmetric flows. SIP and SDP carry IP addresses and ports embedded in the packet and standard NAT implementations only modify the IP and TCP/UDP headers. A true SIP ALG is required to modify the packets as needed for NAT, but also to open holes in the firewall as needed for traffic flow based on the information carried in the SIP header.

Enabling the AOS SIP ALG (using the **ip firewall alg sip** command) configures the firewall to examine the ALL SIP packets it identifies and maintain knowledge of SIP transmissions on the network. Since SIP packet headers include port information for the call setup, the ALG must intelligently read the packets and remember the information. To accomplish this, the SIP ALG enables two other SIP functions, the SIP stack (**ip sip** command) and the SIP proxy server (**ip sip proxy**). This operation allows dynamic configuration of the SIP network, because UAs on the network do not need to be manually added to the router's location database. If there is a SIP node on the network that transmits traffic, the router will identify the traffic as SIP traffic and maintain the appropriate information. This mode can be considered "transparent-proxy." An ADTRAN router running in transparent-proxy mode can be added to a previously configured network (without requiring specific SIP location database configuration) and can be expected to intelligently route SIP packets.

As an alternative to running in transparent-proxy mode, the AOS SIP proxy server can be configured to restrict SIP knowledge to only nodes entered into the location database (using the **no ip firewall alg sip** command). Just as a router uses an IP route table to determine the destination for IP packets it receives, a SIP proxy server uses the location database to determine the appropriate destination UA. Manually configuring the location database can be cumbersome for a large SIP network. To avoid losing pertinent information in the event of a power loss, use the **ip sip database local** command to create a persistent database on the local router memory that is maintained across a power loss.

# ip firewall attack-log threshold <*value*>

Use the **ip firewall attack-log threshold** command to specify the number of attack mounting attempts the AOS will identify before generating a log message. Use the **no** form of this command to return to the default threshold.

> NOTE *The AOS security features must be enabled (using the **ip firewall** command) for the stateful inspection firewall to be activated.*

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the number of attack mounting attempts the AOS will identify before generating a log message. Valid range: 0 to 4,294,967,295. |

## Default Values

By default, the **ip firewall attack-log threshold** is set at 100.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |

## Usage Examples

The following example specifies a threshold of 25 attacks before generating a log message:

(config)**#ip firewall attack-log threshold 25**

# ip firewall check reflexive-traffic

Use the **ip firewall check reflexive-traffic** command to enable the AOS stateful inspection firewall to process traffic from a primary subnet to a secondary subnet on the same interface through the firewall. Use the **no** form of this command to disable this feature.

*The AOS security features must be enabled (using the **ip firewall** command) for the stateful inspection firewall to be activated.*

## Syntax Description

No subcommands.

## Default Values

All AOS security features are disabled by default until the **ip firewall** command is issued at the Global Configuration prompt. In addition, the reflexive traffic check is disabled until the **ip firewall check reflexive-traffic** command is issued.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Functional Notes

This command allows the firewall to process traffic from a primary subnet to a secondary subnet on the same interface through the firewall. If enabled, this traffic will be processed through the access policy on that interface and any actions specified will be executed on the traffic.

## Usage Examples

The following example enables the AOS reflexive traffic check:

(config)**#ip firewall check reflexive-traffic**

# ip firewall check rst-seq

Use the **ip firewall check rst-seq** command to enable TCP reset sequence number checking. Use the **no** form of this command to disable this feature.

*The AOS security features must be enabled (using the **ip firewall** command) for the stateful inspection firewall to be activated.*

## Syntax Description

No subcommands.

## Default Values

All AOS security features are disabled by default until the **ip firewall** command is issued at the Global Configuration prompt. In addition, TCP reset sequence number checking is disabled until the **ip firewall check rst-seq** command is issued.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Usage Examples

The following example enables TCP reset sequence number checking:

(config)**#ip firewall check rst-seq**

# ip firewall check syn-flood

Use the **ip firewall check syn-flood** command to enable the AOS stateful inspection firewall to filter out phony TCP service requests and allow only legitimate requests to pass through. Use the **no** form of this command to disable this feature.


*The AOS security features must be enabled (using the **ip firewall** command) for the stateful inspection firewall to be activated.*

## Syntax Description

No subcommands.

## Default Values

All AOS security features are inactive until the **ip firewall** command is issued at the Global Configuration prompt. In addition, the SYN-flood check is enabled by default but remains inactive until the **ip firewall** command is issued.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |

## Functional Notes

SYN flooding is a well-known denial of service attack on TCP-based services. TCP requires a three-way handshake before actual communications begin between two hosts. A server must allocate resources to process new connection requests that are received. A potential intruder is capable of transmitting large amounts of service requests (in a very short period of time), causing servers to allocate all resources to process the phony incoming requests. Using the **ip firewall check syn-flood** command configures the AOS stateful inspection firewall to filter out phony service requests and allow only legitimate requests to pass through.

## Usage Examples

The following example disables the AOS SYN-flood check:

(config)**#no ip firewall check syn-flood**

# ip firewall check winnuke

Use the **ip firewall check winnuke** command to enable the AOS stateful inspection firewall to discard all out of band (OOB) data (to protect against WinNuke attacks). Use the **no** form of this command to disable this feature.


*The AOS security features must be enabled (using the **ip firewall** command) for the stateful inspection firewall to be activated.*

## Syntax Description

No subcommands.

## Default Values

All AOS security features are inactive until the **ip firewall** command is issued at the Global Configuration prompt. In addition, WinNuke attack checking is disabled until the **ip firewall check winnuke** command is issued.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |

## Functional Notes

WinNuke attack is a well-known denial of service attack on hosts running Microsoft Windows® operating systems. An intruder sends out of band (OOB) data over an established connection to a Windows user. Windows cannot properly handle the OOB data and the host reacts unpredictably. Normal shut-down of the hosts will generally return all functionality. Using the **ip firewall check winnuke** command configures the AOS stateful inspection firewall to filter all OOB data to prevent network problems.

## Usage Examples

The following example enables the firewall to filter all OOB data:

(config)**#ip firewall check winnuke**

# ip firewall fast-nat-failover

Use the **ip firewall fast-nat-failover** command to delete associations on default route policy-class changes. Use the **no** form of this command to disable this feature.

*The AOS security features must be enabled (using the **ip firewall** command) for the stateful inspection firewall to be activated.*

## Syntax Description

No subcommands.

## Default Values

All AOS security features are disabled by default until the **ip firewall** command is issued at the Global Configuration prompt. In addition, the fast NAT failover is disabled until the **ip firewall fast-nat-failover** command is issued.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example enables **fast-nat-failover**:

(config)**#ip firewall fast-nat-failover**

# ip firewall fin-timeout <*value*>

Use the **ip firewall fin-timeout** command to specify the time period allowed for Transport Control Protocol (TCP) FIN. Use the **no** form of this command to return to the default setting.

*The AOS security features must be enabled (using the **ip firewall** command) for the stateful inspection firewall to be activated.*

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the time period in seconds allowed for TCP FIN. Range is 0 to 4,294,967,295 seconds. |

## Default Value

By default, **ip firewall fin-timeout** is set to 4 seconds.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example sets the TCP FIN time period to 120 seconds:

(config)**#ip firewall fin-timeout 120**

# ip firewall policy-log threshold *<value>*

Use the **ip firewall policy-log threshold** command to specify the number of connections required by an access control policy before the AOS will generate a log message. Use the **no** form of this command to return to the default threshold.

*The AOS security features must be enabled (using the **ip firewall** command) for the stateful inspection firewall to be activated.*

## Syntax DescriptionSyntax Description

| | |
|---|---|
| *<value>* | Specifies the number of access policy connections the AOS will identify before generating a log message. Valid range: 0 to 4,294,967,295. |

## Default Values

By default, the **ip firewall policy-log threshold** is set to 100.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |

## Usage Examples

The following example specifies a threshold of 15 connections before generating a log message:

(config)**#ip firewall policy-log threshold 15**

# ip firewall rst-timeout <*value*>

Use the **ip firewall rst-timeout** command to specify the time period allowed for Transport Control Protocol (TCP) reset. Use the **no** form of this command to return to the default setting.

*The AOS security features must be enabled (using the **ip firewall** command) for the stateful inspection firewall to be activated.*

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the time period in seconds allowed for TCP reset. Range is 0 to 4,294,967,295 seconds. |

## Default Value

By default, **ip firewall rst-timeout** is set to 20 settings.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example sets the TCP reset time period to 120 seconds:

(config)**#ip firewall rst-timeout 120**

# ip firewall stealth

Use the **ip firewall stealth** command to disable TCP reset for denied firewall associations. The stealth setting allows the route to be invisible as a route hop to associated devices.


*The AOS security features must be enabled (using the **ip firewall** command) for the stateful inspection firewall to be activated.*

## Syntax Description

No subcommands.

## Default Values

All AOS security features are disabled by default until the **ip firewall** command is issued at the Global Configuration prompt. In addition, the stealth option is disabled until the **ip firewall stealth** command is issued.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example enables the **stealth** option:

(config)**#ip firewall stealth**

# ip forward-protocol udp <*value*>

Use the **ip forward-protocol udp** command to specify the protocols and ports the AOS allows when forwarding broadcast packets. Use the **no** form of this command to disable a specified protocol or port from being forwarded.

> NOTE *The **ip helper** command must be used in conjunction with the **ip forward-protocol** command to configure the AOS to forward UDP broadcast packets. Refer to ip ffe on page 842 for more information.*

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the UDP traffic type (using source port). |

The following is the list of UDP port numbers that may be identified using the text name:

| | |
|---|---|
| **biff** (Port 512) | **pim-auto-rp** (Port 496) |
| **bootps** (Port 67) | **rip** (Port 520) |
| **discard** (Port 9) | **snmp** (Port 161) |
| **dnsix** (Port 195) | **snmptrap** (Port 162) |
| **domain** (Port 53) | **sunrpc** (Port 111) |
| **echo** (Port 7) | **syslog** (Port 514) |
| **isakmp** (Port 500) | **tacacs** (Port 49) |
| **mobileip** (Port 434) | **talk** (Port 517) |
| **nameserver** (Port 42) | **tftp** (Port 69) |
| **netbios-dgm** (Port 138) | **time** (Port 37) |
| **netbios-ns** (Port 137) | **who** (Port 513) |
| **netbios-ss** (Port 139) | **xdmcp** (Port 177) |
| **ntp** (Port 123) | |

Alternately, the <*value*> may be specified using the following syntax: <0-65535>. Specifies the port number used by UDP to pass information to upper layers. All ports below 1024 are considered well-known ports and are controlled by the Internet Assigned Numbers Authority (IANA). All ports above 1024 are dynamically assigned ports that include registered ports for vendor-specific applications.

## Default Values

By default, the AOS forwards broadcast packets for all protocols and ports.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |

## Functional Notes

Use this command to configure the AOS to forward UDP packets across the WAN link to allow remote devices to connect to a UDP service on the other side of the WAN link.

## Usage Examples

The following example forwards all Domain Name Server (DNS) broadcast traffic to the DNS server with IP address 192.33.5.99:

(config)**#ip forward-protocol udp domain**
(config)**#interface eth 0/1**
(config-eth 0/1)**#ip helper-address 192.33.5.99**

# ip ftp access-class <*name*> in

Use the **ip ftp access-class in** command to assign an access policy to all self-bound File Transfer Protocol (FTP) sessions. Use the **no** form of this command to disable this feature.

## Syntax Description

| | |
|---|---|
| <*name*> | Specifies the configured access policy (ACP) to apply to inbound FTP traffic. |

## Default Values

By default, all FTP access is allowed.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |

## Usage Examples

The following example applies the configured ACP (labeled **Inbound_FTP**) to inbound FTP traffic:

(config)**#ip ftp access-class Inbound_FTP in**

# ip ftp source-interface <*interface*>

Use the **ip ftp source-interface** command to use the specified interface's IP address as the source IP address for FTP traffic transmitted by the unit. Use the **no** form of this command if you do not wish to override the normal source IP address.

## Syntax Description

| | |
|---|---|
| <*interface*> | Specifies the interface to be used as the source IP address for FTP traffic. Specify an interface in the format <*interface type [slot/port | slot/port.sub-interface id | interface id | interface id.sub-interface id]*>. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **ip ftp source-interface?** for a complete list of valid interfaces. |

## Default Values

No default value is necessary for this command.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |
| Release 9.1 | Command expanded to include HDLC interface. |

## Functional Notes

This command allows you to override the *sender* field in the IP packet. If you have multiple interfaces in your unit, changing the *sender* tells the receiver where to send replies. This functionality can also be used to allow packets to get through firewalls that would normally block the flow.

## Usage Examples

The following example configures the unit to use the **loopback 1** interface as the source IP for FTP traffic:

(config)**#ip ftp source-interface loopback 1**

# ip host *<name> <ip address>*

Use the **ip host** command to define an IP host name. This allows you to statically map host names and addresses in the host cache. Use the **no** form of this command to remove defined maps.

## Syntax Description

| | |
|---|---|
| *<name>* | Defines the name of the host. |
| *<ip address>* | Specifies IP address associated with this IP host. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

## Default Values

By default, the host table is empty.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Functional Notes

The name may be any combination of numbers and letters as long as it is not a valid IP address or does not exceed 256 characters.

## Usage Examples

The following example defines two static mappings:

(config)**#ip host mac 10.2.0.2**
(config)**#ip host dal 172.38.7.12**

# ip http

Use the **ip http** command to enable Web access to the unit. Use the **no** form of these commands to disable these features. Variations of this command include:

**ip http access-class** <*name*> **in**
**ip http authentication** <*name*>
**ip http secure-access-class** <*name*> **in**
**ip http secure-server**
**ip http secure-server** <*name*>
**ip http server**
**ip http server** <*name*>
**ip http session-limit** <*number*>
**ip http session-timeout** <*name*>

## Syntax Description

| | |
|---|---|
| **access-class** <*name*> | Restricts access to the HTTP server using the specified access control list. |
| **in** | Applies to all incoming connections. |
| **authentication** <*name*> | Assigns the specified AAA list to HTTP authentication. |
| **secure-access-class** <*name*> | Restricts access to the HTTPS server using the specified secure access control list. |
| **secure-server** <*name*> | Enables the specified SSL server. |
| **server** <*name*> | Enables the specified HTTP server connection. |
| **session-limit** <*number*> | Sets the maximum number of sessions allowed. Valid range is 0 to 100 with 100 as the default. |
| **session-timeout** <*value*> | Sets the session timeout value. Valid range is 10 to 86,400 seconds. The default is 600. |

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Usage Examples

The following example enables Web access to the router:

(config)**#ip http server**

# ip igmp join <*ip address*>

Use the **ip igmp join** command to instruct the router stack to join a specific group. The stack may join multiple groups. Use the **no** form of this command to disable this feature.

## Syntax Description

| | |
|---|---|
| <*ip address*> | Specifies the IP address of a multicast group. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

## Default Values

No defaults necessary for this command.

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |

## Functional Notes

This command aids in debugging, allowing the router's IP stack to connect to and respond on a multicast group. The local stack operates as an IGMP host on the attached segment. In multicast stub applications, the global helper address takes care of forwarding IGMP joins/responses on the upstream interface. The router may respond to ICMP echo requests for the joined groups.

## Usage Examples

The following example configures the unit to join with the specified multicast group:

(config)**#ip igmp join 172.0.1.50**

# ip igmp snooping

Use the **ip igmp snooping** command to globally enable Internet Group Management Protocol (IGMP) snooping. Use the **no** form of this command to disable global IGMP snooping.

## Syntax Description

No subcommands.

## Default Values

By default, IGMP snooping is enabled.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Functional Note

IGMP Snooping is a method of preventing switches from flooding all ports with received multicast streams. By monitoring the conversations between a host and a router, the switch can determine which multicast streams will interest a host and load its own forwarding tables to take advantage of that knowledge. When the host sends a leave message to the router, the switch removes the entries after a timeout period.

## Usage Examples

The following example globally enables IGMP snooping:

(config)**#ip igmp snooping**

# ip igmp snooping vlan *<vlan id>*

Use the **ip igmp snooping vlan** command to enable Internet Group Management Protocol (IGMP) snooping on the specified virtual local area network (VLAN). Use the **no** form of this command to disable VLAN IGMP snooping.


*Global IGMP snooping overrides the VLAN IGMP snooping. If global snooping is disabled, you cannot enable VLAN IGMP snooping. If global snooping is enabled, you can enable or disable VLAN IGMP snooping. Refer to ip igmp snooping* *on page 539* *for more information*

## Syntax Description

| | |
|---|---|
| *<vlan id>* | Specifies a valid VLAN interface ID. Range is 1 to 4095. |

## Default Values

By default, VLAN IGMP snooping is enabled.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example enables IGMP snooping on VLAN 1:

(config)**#ip igmp snooping vlan 1**

# ip igmp snooping vlan <*vlan id*> mrouter interface <*interface*>

Use the **ip igmp snooping vlan mrouter interface** command to add a static connection to a multicast router. Use the **no** form of this command to remove a static connection to a multicast router.

## Syntax Description

| | |
|---|---|
| <*vlan id*> | Specifies a valid VLAN interface ID. Range is 1 to 4094. |
| <*interface*> | Specifies an interface to be added to the multicast router. Specify an interface in the format <*interface type [slot/port | slot/port.sub-interface id | interface id | interface id.sub-interface id]*>. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **ip igmp snooping vlan** <*vlan id*> **mrouter interface ?** for a complete list of applicable interfaces. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example adds Ethernet interface 0/1 to the list of multicast router interfaces:

(config)**#ip igmp snooping vlan 1 mrouter interface ethernet 0/1**

# ip igmp snooping vlan <*vlan id*> static <*mac address*> interface <*interface*>

Use the **ip igmp snooping vlan static interface** command to statically configure a Layer 2 interface as a member of a multicast group. Use the **no** form of this command to remove a Layer 2 interface from a multicast group.

## Syntax Description

| | |
|---|---|
| <*vlan id*> | Specifies the VLAN ID of the multicast group. Range is 1 to 4094. |
| <*mac address*> | Specifies the group's 48-bit MAC address. MAC addresses should be expressed in following format xx:xx:xx:xx:xx:xx (for example, **00:A0:C8:00:00:01**). |
| <*interface*> | Specifies an interface identification for the member interface. Specify an interface in the format <*interface type [slot/port | slot/port.sub-interface id | interface id | interface id.sub-interface id]*>. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **ip igmp snooping vlan** <*vlan id*> **static** <*mac address*> **interface ?** for a complete list of applicable interfaces. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Functional Notes

There are two types of multicast addresses: MAC addresses and IP addresses. A multicast IP address is a Class D address (224.0.0.0 to 239.255.255.255). These addresses are also referred to as Group Destination Addresses (GDA). Each GDA has an associated multicast MAC address. A multicast MAC address is formed by using the prefix 01-00-5e followed by the last 23 bits of the GDA. The <*mac address*> specified in this command must be a multicast MAC address. The following table shows examples of multicast MAC addresses.

**Multicast Addresses**

| Multicast IP Address | Multicast MAC Address |
|---|---|
| 226.10.10.10 | 01-00-5e-0a-0a-0a |
| 228.20.20.20 | 01-00-5e-14-14-14 |
| 230.30.30.30 | 01-00-5e-1e-1e-1e |

This mapping of IP addresses is a many-to-one relationship. For example, 226.10.10.10 maps to the same MAC address as 227.10.10.10. The entire Class D network is not available for multicast. The following table shows the reserved addresses.

**Reserved Multicast IP Addresses**

| | |
|---|---|
| 224.0.0.1 | All Multicast-capable hosts |
| 224.0.0.2 | All Multicast-capable routers |
| 224.0.0.5 and 224.0.0.6 | Reserved for OSPF |
| 224.0.0.1 to 224.0.0.255 | Generally reserved for various protocols |

## Usage Examples

The following example configures the Ethernet interface 0/1 as a member of the multicast group with multicast MAC address **01:00:5E:01:01:01**:

(config)**#ip igmp snooping vlan 1 static 01:00:5E:01:01:01 interface ethernet 0/1**

# ip load-sharing

Use the **ip load-sharing** command to configure whether parallel routes in the route table are used to load-share forwarded packets. If this command is disabled, the route table uses a single "best" route for a given subnet. If this command is enabled, the route table can use multiple "best" routes and alternate between them. Use the **no** form of this command to disable this feature. Variations of this command include:

**ip load-sharing per-destination**
**ip load-sharing per-packet**

## Syntax Description

| | |
|---|---|
| **per-destination** | Specifies that the route used for forwarding a packet be based on a hash of the source and destination IP address in the packet. |
| **per-packet** | Specifies that each forwarding route lookup rotates through all the parallel "best" routes. (Parallel routes are defined as routes to the same subnet with the same metrics that only differ by their next hop address.) |

## Default Values

By default, ip load-sharing is disabled.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example turns on load-sharing per destination:

(config)**#ip load-sharing per-destination**

The following example disables load-sharing:

(config)**#no ip load-sharing**

# ip local policy route-map *<name>*

Use the **ip local policy route-map** command to specify a route map for local policy routing on the device. This setting is applied to the local network interface. Use the **no** form of this command to return to the default route map.

## Syntax Description

| | |
|---|---|
| *<name>* | Specify the name of the route map. |

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Functional Notes

Before a route map can be specified, it must first be defined using the **route-map** command. Refer to *route-map* on page 650 for more information.

## Usage Examples

The following example specifies a route map entitled **myMap** for local policy routing:

(config)**#ip local policy route-map myMap**

# ip mcast-stub helper-address *<ip address>*

Use the **ip mcast-stub helper-address** command to specify an IP address toward which IGMP host reports and leave messages are forwarded. This command is used in IP multicast stub applications in conjunction with the **ip mcast-stub downstream** and **ip mcast-stub upstream** commands. Use the **no** form of this command to return to default.

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies the address to which the IGMP host reports and leave messages are forwarded. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

## Default Values

By default, no helper-address is configured.

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |

## Functional Notes

The helper address is configured globally and applies to all multicast-stub downstream interfaces. The address specified may be the next upstream hop or any upstream address on the distribution tree for the multicast source, up to and including the multicast source. The router selects, from the list of multicast-stub upstream interfaces, the interface on the shortest path to the specified address. The router then proxies, on the selected upstream interface (using an IGMP host function), any host joins/leaves received on the downstream interface(s). The router retransmits these reports with addresses set as if the report originated from the selected upstream interface.

For example, if the router receives multiple joins for a group, it will not send any extra joins out the upstream interface. Also, if it receives a leave, it will not send a leave until it is certain that there are no more subscribers on any downstream interface.

## Usage Examples

The following example specifies 172.45.6.99 as the helper address:

(config)**#ip mcast-stub helper-address 172.45.6.99**

# ip multicast-routing

Use the **ip multicast-routing** command to enable the multicast router process. The command does not affect other multicast-related configurations. Use the **no** form of this command to disable this feature. Disabling this command prevents multicast forwarding but does not remove other multicast commands and processes.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |

## Usage Examples

The following example enables multicast functionality:

(config)**#ip multicast-routing**

# ip name-server

Use the **ip name-server** command to designate one or more name servers to use for name-to-address resolution. Use the **no** form of this command to remove any addresses previously specified. Variations of this command include:

**ip name-server** *<ip address1>*
**ip name-server** *<ip address1> <ip address2>*
**ip name-server** *<ip address1> <ip address2> <ip address3>*
**ip name-server** *<ip address1> <ip address2> <ip address3> <ip address4>*
**ip name-server** *<ip address1> <ip address2> <ip address3> <ip address4> <ip address5>*
**ip name-server** *<ip address1> <ip address2> <ip address3> <ip address4> <ip address5> <ip address6>*

## Syntax Description

| | |
|---|---|
| *<ip address 1-6>* | Specifies up to six name-server addresses. |

## Default Values

By default, no name servers are specified.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Usage Examples

The following example specifies host 172.34.1.111 as the primary name server and host 172.34.1.2 as the secondary server:

(config)**#ip name-server 172.341.1.111 172.34.1.2**

This command will be reflected in the configuration file as follows:

ip name-server 172.34.1.111 172.34.1.2

# ip policy-class *<name>*

Use the **ip policy-class** command to create an access control policy and enter the access control policy command set. Use the **no** form of this command to delete an access policy and all the entries contained in it.Variations of this command include:

**ip policy-class** *<policyname>*
*<action>*


*Configured access policies will only be active if the **ip firewall** command has been entered at the Global Configuration mode prompt to enable the AOS security features. All configuration parameters are valid, but no security data processing will be attempted unless the security features are enabled.*


*Before applying an access control policy to an interface, verify your Telnet connection will not be affected by the policy. If a policy is applied to the interface you are connecting through and it does not allow Telnet traffic, your connection will be lost.*

## Syntax Description

| | |
|---|---|
| *<name>* | Identifies the configured access policy using an alphanumeric descriptor (maximum of 255 characters). All access policy descriptors are case-sensitive. |
| *<action>* | **allow list** *<access control list name>* **[policy** *<access policy name>* **\| self] [stateless]**<br>All packets permitted by the access control list (ACL) will be allowed to enter the interface to which the policy class is assigned and an association will be created in the firewall. All associations created by the **allow list** are subject to the built-in firewall timers (refer to *ip policy-timeout <protocol> <range> <port> <value>* on page 557). All packets denied by the ACL will be processed by the next policy class entry or implicitly discarded if no further policy class entries exist.<br>When the **policy** *<access policy name>* is specified, the firewall attempts to match the specified access policy with the access policy that is applied to the packet's egress interface as determined by the routing table or policy-based routing configuration. If there is a match, the firewall will process the packet. If there is no match, the firewall will process the packet based on the next policy class entry or implicitly discard it if no further policy class entries exist.<br>When the **self** keyword is applied, packets permitted by the ACL destined for any local interface on the unit will be allowed. These packets are terminated by the unit and are not routed or forwarded to other destinations. Using the **self** keyword is helpful when opening up remote administrative access to the unit (Telnet, SSH, ICMP, Web GUI). |

When the **stateless** keyword is applied, traffic is not subject to the built-in firewall timers. Stateless traffic bypasses the application-level gateways (ALGs). Stateless processing is helpful when passing traffic over VPN tunnels. Traffic sent over VPN tunnels is purposely selected and encrypted; there is no need to firewall the traffic as well. VPN configurations created using the VPN Wizard in the Web GUI use **stateless** processing by default.

**allow reverse list** <*access control list name*> **[policy** <*access policy name*> **| self] [stateless]**

The **allow reverse list** command is identical in function to the **allow list** command with the exception of the **reverse** keyword. The **reverse** keyword instructs the firewall to use the source information as the destination information and visa versa in the specified ACL.

**discard list** <*access control list name*> **[policy** <*access policy name*> **| self]**

All packets permitted by the ACL will be explicitly discarded upon entering the interface that the policy class is assigned to. All packets denied by the ACL will be processed by the next policy class entry or implicitly discarded if no further policy class entries exist.

When the **policy** <*access policy name*> is specified, the firewall attempts to match the specified access policy with the access policy that is applied to the packet's egress interface as determined by the routing table or policy-based routing configuration. If there is a match, the firewall will process the packet. If there is no match, the firewall will process the packet based on the next policy class entry or implicitly discard it if no further policy class entries exist.

When the **self** keyword is applied, packets permitted by the access-control list destined for any local interface on the unit will be implicitly discarded.

**nat source list** <*access control list name*> **address** <*address*> **overload [policy** <*access policy name*>**]**

or

**nat source list** <*access control list name*> **interface** <*interface*> **overload [policy** <*access policy name*>**]**

All packets permitted by the ACL entering the interface to which the policy class is assigned will translate the source IP address of the packet to the specified **address** or **interface** and an association will be created in the firewall. The **address** keyword specifies the IP address from which the translated packets will be sourced. The primary IP address of an interface is used as the source IP for translated packets when the **interface** keyword is applied. This function is commonly referred to as a “many-to-one NAT”. All

associations created by the **nat source list** are subject to the built-in firewall timers (refer to *ip policy-timeout <protocol> <range> <port> <value>* on page 557). All packets denied by the extended access control list will be processed by the next policy class entry or implicitly discarded if no further policy class entries exist. The **overload** command is not optional and must be used when using the **nat source list** command.

When the **policy** <*access policy name*> is specified, the firewall attempts to match the specified access policy with the access policy that is applied to the packet's egress interface as determined by the routing table or policy-based routing configuration. If there is a match, the firewall will process the packet. If there is no match, the firewall will process the packet based on the next policy class entry or implicitly discard it if no further policy class entries exist.

**nat destination list** <*extended access control list name*> **address** <*ip address*> **[port** <*port*>**]**

All packets permitted by the specified extended ACL entering the interface that the policy class is assigned to will translate the destination IP address of the packet to the specified **address** and an association will be created in the firewall. The **address** keyword specifies the private IP host to which the translated packets are destined. All associations created by the **nat destination list** are subject to the built-in firewall timers (refer to *ip policy-timeout <protocol> <range> <port> <value>* on page 557). All packets denied by the extended ACL will be processed by the next policy class entry or implicitly discarded if no further policy class entries exist. The **port** keyword is used to translate the original destination port to a user-specified port.

## Default Values

By default, all AOS security features are disabled and there are no configured access lists.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |

## Functional Notes

AOS access control policies are used to allow, discard, or manipulate (using **NAT**) data for each physical interface. Each ACP consists of a selector (**access list**) and an action (**allow**, **discard**, **NAT**). When packets are received on an interface, the configured ACPs are applied to determine whether the data will be processed or discarded.


*An implicit discard exists at the end of every policy class. Specifying a **discard list** is unnecessary in most applications and should be used with caution. A **discard list** can adversely affect certain functions of a unit (VPN, routing protocols, etc.). Specifying an empty ACL or a non-existent ACL in a policy class will result in an implicit permit.*

## Usage Examples

The following is an example of adding policy class entries (ACL self and ACL MATCHALL) to a policy class named **Private**:

(config)**#ip policy-class Private**
(config-policy-class)**#allow list self self**
(config-policy-class)**#nat destination list MATCHALL interface ppp 1 overload**

The following is a sample output of the configuration after issuing these commands:

```
!
ip access-list standard wizard-ics
 remark Internet Connection Sharing
 permit any
!
ip access-list extended self
 remark Traffic to NetVanta
 permit ip any any log
!
ip policy-class Private
 allow list self self
 nat source list wizard-ics interface ppp 1 overload
!
```

## Technology Review

Creating access policies and lists to regulate traffic through the routed network is a four-step process:

Step 1:
Enable the security features of the AOS using the **ip firewall** command.

Step 2:
Create an access control list to permit or deny specified traffic. Standard ACLs match based on the source of the packet. Extended ACLs match based on the source and destination of the packet. Sources can be expressed in one of four ways:

1. Using the keyword **any** to match any IP address.
2. Using **host** <*A.B.C.D*> to specify a single host address.
3. Using the <*A.B.C.D*> <*wildcard*> format to match all IP addresses in a range. Wildcard masks work in reverse logic from subnet masks. Specifying 255 in any octet of the wildcard mask equates to a “don’t care”.
4. Using the keyword **hostname** to match based on a DNS name. The unit must be configured with DNS servers for this function to work.

Step 3:
Create an access policy that uses a configured access list. AOS access policies are used to allow, discard, or manipulate (using **NAT**) data for each physical interface. Each ACP consists of a selector (**access list**) and an action (**allow**, **discard**, **NAT**). When packets are received on an interface, the configured ACPs are applied to determine whether the data will be processed or discarded. Possible actions performed by the access policy are as follows:

**allow list** <*access control list name*>
**allow list** <*access control list name*> **stateless**
**allow list** <*access control list name*> **policy** <*access policy name*>
**allow list** <*access control list name*> **policy** <*access policy name*> **stateless**
**allow list** <*access control list name*> **self**
**allow list** <*access control list name*> **self stateless**
**discard list** <*access control list name*>
**discard list** <*access control list name*> **policy** <*access policy name*>
**discard list** <*access control list name*> **self**
**nat destination list** <*access control list name*> **address** <*IP address*> **port** <*port number*>
**nat source list** <*access control list name*> **address** <*IP address*> **overload**
**nat source list** <*access control list name*> **address** <*IP address*> **policy** <*access policy name*>
**nat source list** <*access control list name*> **interface** <*interface*> **overload**
**nat source list** <*access control list name*> **interface** <*interface*> **policy** <*access policy name*>

Step 4:
Apply the created access policy to an interface. To assign an access policy to an interface, enter the interface configuration mode for the desired interface and enter **access policy** <*policy name*>. The following example assigns access policy **MatchAll** to the Ethernet 0/1 interface:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#access-policy MatchAll**

# ip policy-class

Use the **ip policy-class max-sessions** and **ip policy-class max-host-sessions** commands to create or alter settings for an access control policy. For more details on IP policy class functionality in AOS, refer to *ip policy-class <name>* on page 549. Use the **no** form of this command to delete an access policy and all the entries contained in it. Variations of this command include the following:

**ip policy-class max-sessions** <*number*>
**ip policy-class** <*name*> **max-host-sessions** <*number*>
**ip policy-class** <*name*> **max-sessions** <*number*>

## Syntax Description

| | |
|---|---|
| <*name*> | Identifies the configured access policy using an alphanumeric descriptor (maximum of 255 characters). All access policy descriptors are case-sensitive. |
| **max-sessions** <*number*> | Specifies the maximum number of allowed policy sessions. Identifying a policy name sets the session limit only for the named policy. Using this command without specifying a policy name sets the limit for the total number of allowed sessions for all policies on the device.<br>This number must be within the appropriate range limits. The limits are either 1 to 4000 or 1 to 30,000 (depending on the type of AOS device you are using). Setting this value to zero turns the feature off. |
| **max-host-sessions** <*number*> | Specifies the maximum number of allowed policy sessions which can be created from each unique source address. This command is used in conjunction with a named policy and only applies the limit to that particular policy.<br>The number must be within the appropriate range limits. The limits are either 1 to 4000 or 1 to 30,000 (depending on the type of AOS device you are using). Setting this value to 0 turns the feature off. |

## Default Values

By default, all AOS security features are disabled and there are no configured access lists.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |
| Release 12.1 | Command was expanded. |

## Usage Examples

The following example allows no more than 100 policy sessions to be sourced from a single host IP address on the **Private** policy class:

(config)**#ip policy-class Private max-host-sessions 100**

The number of maximum sessions can be set with or without a policy name.

The following example sets a total global limit of 55,700 policy sessions allowed on all policy classes:

(config)**#ip policy-class max-sessions 55700**

The following example allows no more than 100 policy sessions on the **Private** policy class:

(config)**#ip policy-class Private max-sessions 100**

The following example removes the policy sessions limit on the **Private** policy class:

(config)**#no ip policy-class Private max-sessions 100**

# ip policy-class *<name>* rpf-check

Use the **ip policy-class rpf-check** command to verify that traffic has entered on the appropriate interface using a route lookup. Reverse Path Forwarding (RPF) is essentially a spoofing check. For more details on IP policy class functionality in AOS, refer to *ip policy-class <name>* on page 549. Use the **no** form of this command to disable this feature.

## Syntax Description

| | |
|---|---|
| *<name>* | Identifies the configured access policy using an alphanumeric descriptor (maximum of 255 characters). All access policy descriptors are case-sensitive. |
| **rpf-check** | Enables RPF check (spoofing). |

## Default Values

This command is enabled by default.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Functional Notes

The **rpf-check** feature should be disabled if your application allows incoming traffic on policy classes that do not match the route table's source destination specifications. This feature can be disabled on a per policy class basis by issuing the command in conjunction with the policy class name you do not want to be checked.

## Usage Examples

The following example turns off the **rpf-check** feature for the **Private** policy class:

(config)**#no ip policy-class Private rpf-check**

# ip policy-timeout *<protocol> <range> <port> <value>*

Use multiple **ip policy-timeout** commands to customize timeout intervals for protocols (TCP, UDP, ICMP, AHP, GRE, ESP) or specific services (by listing the particular port number). Use the **no** form of this command to return to the default timeout values.

## Syntax Description

| | |
|---|---|
| *<protocol>* | Specifies the data protocol such as ICMP, TCP, UDP, AHP, GRE, or ESP. |
| *<range>* | Optional. Customizes timeout intervals for a range of TCP or UDP ports. |
| *<port>* | Specifies the service port to apply the timeout value to; valid only for specifying TCP and UDP services (not allowed for ICMP). |

The following is the list of TCP port numbers that may be identified using the text name (in **bold**):

| | |
|---|---|
| **all-ports** | **kshell** (Port 544) |
| **bgp** (Port 179) | **login** (Port 513) |
| **chargen** (Port 19) | **lpd** (Port 515) |
| **cmd** (Port 514) | **nntp** (Port 119) |
| **daytime** (Port 13) | **pim-auto-rp** (Port 496) |
| **discard** (Port 9) | **pop2** (Port 109) |
| **domain** (Port 53) | **pop3** (Port 110) |
| **echo** (Port 7) | **smtp** (Port 25) |
| **exec** (Port 512) | **ssh** (Port 22) |
| **finger** (Port 79) | **sunrpc** (Port 111) |
| **ftp** (Port 21) | **syslog** (Port 514) |
| **ftp-data** (Port 20) | **tacacs** (Port 49) |
| **gopher** (Port 70) | **talk** (Port 517) |
| **hostname** (Port 101) | **telnet** (Port 23) |
| **https** (443) | **time** (Port 37) |
| **ident** (Port 113) | **uucp** (Port 540) |
| **irc** (Port 194) | **whois** (Port 43) |
| **klogin** (Port 543) | **www** (Port 80) |

The following is the list of UDP port numbers that may be identified using the text name (in **bold**):

| | |
|---|---|
| **all-ports** | **ntp** (Port 123) |
| **biff** (Port 512) | **pim-auto-rp** (Port 496) |
| **bootpc** (Port 68) | **rip** (Port 520) |
| **bootps** (Port 67) | **snmp** (Port 161) |
| **discard** (Port 9) | **snmptrap** (Port 162) |
| **dnsix** (Port 195) | **sunrpc** (Port 111) |
| **domain** (Port 53) | **syslog** (Port 514) |
| **echo** (Port 7) | **tacacs** (Port 49) |

| | |
|---|---|
| | **isakmp** (Port 500) **talk** (Port 517)<br>**mobile-ip** (Port 434) **tftp** (Port 69)<br>**nameserver** (Port 42) **time** (Port 37)<br>**netbios-dgm** (Port 138) **who** (Port 513)<br>**netbios-ns** (Port 137) **xdmcp** (Port 177)<br>**netbios-ss** (Port 139) |
| <*value*> | Specifies the wait interval (in seconds) before an active session is closed. Valid range is 0 to 4,294,967,295 seconds. |

## Default Values

| | |
|---|---|
| <*value*> | The following default policy timeout intervals values apply:<br>**tcp** (600 seconds; 10 minutes)<br>**udp** (60 seconds; 1 minute)<br>**icmp** (60 seconds; 1 minute)<br>**ahp** (60 seconds; 1 minute)<br>**gre** (60 seconds; 1 minute)<br>**esp** (60 seconds; 1 minute) |

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |
| Release 11.1 | Added AHP, GRE, and ESP policies. |

## Usage Examples

The following example creates customized policy timeouts for the following:

Internet traffic (TCP Port 80) timeout 24 hours (86400 seconds)
Telnet (TCP Port 23) timeout 20 minutes (1200 seconds)
FTP (21) timeout 5 minutes (300 seconds)
All other TCP services timeout 8 minutes (480 seconds)

(config)**#ip policy-timeout tcp www 86400**
(config)**#ip policy-timeout tcp telnet 1200**
(config)**#ip policy-timeout tcp ftp 300**
(config)**#ip policy-timeout tcp all_ports 480**

The following example creates customized policy timeouts for UDP NetBIOS ports 137 to 139 of 200 seconds and UDP ports 6000 to 7000 of 300 seconds:

(config)**#ip policy-timeout udp range netbios-ns netbios-ss 200**
(config)**#ip policy-timeout udp range 6000 7000 300**

The following example creates a customized policy timeout of 1200 seconds for ESP:

(config)**#ip policy-timeout esp 1200**

The following example creates a customized policy timeout of 1200 seconds for GRE:

(config)**#ip policy-timeout gre 1200**

The following example creates a customized policy timeout of 1200 seconds for AHP:

(config)**#ip policy-timeout ahp 1200**

# ip prefix-list <*name*> description <*"text"*>

Use the **ip prefix-list description** command to create and name prefix lists. Use the **no** form of this command to remove a prefix list.

## Syntax Description

| | |
|---|---|
| <*name*> | Specifies a particular prefix list. |
| **description** <*"text"*> | Assigns text (enclosed in quotation marks) used as a description for the prefix list. Maximum length is 80 characters. |

## Default Values

No default values are necessary for this command.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Functional Notes

This command adds a string of up to 80 characters as a description for a prefix list. It also creates the prefix list if a prefix list of that name does not already exist.

## Usage Examples

The following example adds a description to the prefix-list **test**:

(config)**#ip prefix-list test description "An example prefix list"**

# ip prefix-list <*name*> seq <*number*>

Use the **ip prefix-list seq** command to specify a prefix to be matched or a range of mask lengths. Use the **no** form of this command to remove a prefix list. Variations of this command include:

**ip prefix-list** <*name*> **seq** <*number*> **deny** <*network ip /length*>
**ip prefix-list** <*name*> **seq** <*number*> **deny** <*network ip /length*> **ge** <*value*>
**ip prefix-list** <*name*> **seq** <*number*> **deny** <*network ip /length*> **le** <*value*>
**ip prefix-list** <*name*> **seq** <*number*> **permit** <*network ip /length*>
**ip prefix-list** <*name*> **seq** <*number*> **permit** <*network ip /length*> **ge** <*value*>
**ip prefix-list** <*name*> **seq** <*number*> **permit** <*network ip /length*> **le** <*value*>

## Syntax Description

| | |
|---|---|
| <*listname*> | Specifies a particular prefix list. |
| <*number*> | Specifies the entry's unique sequence number which determines the processing order. Lower-numbered entries are processed first. Range is 1 to 4,294,967,294. |
| **permit** <*network ip /length*> | Permits access to entries matching the specified network IP address and the corresponding network prefix length (for example, **10.10.10.1 /24**). |
| **deny** <*network ip /length*> | Denies access to entries matching the specified network IP address and the corresponding network prefix length (for example, **10.10.10.1 /24**). |
| **le** <*value*> | Specifies the upper end of the range. Range is 0 to 32. |
| **ge** <*value*> | Specifies the lower end of the range. Range is 0 to 32. |

## Default Values

If no ge or le parameters are specified, an exact match is assumed. If only ge is specified, the range is assumed to be from ge-value to 32. If only le is specified, the range is assumed to be from len to le-value.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Functional Notes

This command specifies a prefix to be matched. Optionally, it may specify a range of mask lengths. The following rule must be followed: len < ge-value ≤ le-value. A prefix list with no entries allows all routes. A route that does not match any entries in a prefix list is dropped. As soon as a route is permitted or denied, there is no further processing of the rule in the prefix list. A route that is denied at the beginning entry of a prefix list will not be allowed, even if it matches a permitting entry further down the list.

## Usage Examples

The following example creates a prefix list entry in the prefix list **test** matching only the 10.0.0.0/8 network:

(config)**#ip prefix-list test seq 5 deny 10.0.0.0/8**

The following example creates a prefix list entry in the prefix list **test** matching any network of length 24 or less:

(config)**#ip prefix-list test seq 10 permit 0.0.0.0/0 le 24**

# ip radius source-interface *<interface>*

Use the **ip radius source-interface** command to specify the network-attached storage (NAS) IP address attribute passed with the RADIUS authentication request packet. Use the **no** form of this command to remove a defined source interface.

## Syntax Description

| | |
|---|---|
| *<interface>* | Specifies the source interface. Specify an interface in the format *<interface type [slot/port \| slot/port.sub-interface id \| interface id \| interface id.sub-interface id]>*. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **ip radius source-interface ?** for a complete list of interfaces. |

## Default Values

By default, no source interface is defined.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

If this value is not defined, the address of the source network interface is used.

## Usage Examples

The following example configures the Ethernet 0/1 port to be the source interface:

(config)**#ip radius source-interface ethernet 0/1**

# ip route

Use the **ip route** command to add a static route to the route table. Use the **no** form of this command to remove a configured static route. Variations of this command include:

**ip route** <*ip address*> <*subnet mask*> <*interface*>
**ip route** <*ip address*> <*subnet mask*> <*interface*> <*administrative distance*>
**ip route** <*ip address*> <*subnet mask*> <*ip address*>
**ip route** <*ip address*> <*subnet mask*> <*ip address*> <*administrative distance*>
**ip route** <*ip address*> <*subnet mask*> **null 0**
**ip route** <*ip address*> <*subnet mask*> **null 0** <*administrative distance*>
**ip route** <*ip address*> <*subnet mask*> <*interface*> <*administrative distance*> **track** <*name*>
**ip route** <*ip address*> <*subnet mask*> <*ip address*> <*administrative distance*> **track** <*name*>

## Syntax Description

| | |
|---|---|
| <*ip address*> | Specifies the network address to add to the route table. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| <*subnet mask*> | Specifies the subnet mask that corresponds to a range of IP addresses (network) or a specific host. Subnet masks can be expressed in dotted decimal notation (for example, **255.255.255.0**) or as a prefix length (for example, **/24**). |
| **[**<*interface*> **|** <*ip address*>**]** | Specifies the far end IP address or an egress interface in the unit. Use the **ip route** <*ip address*> <*subnet mask*> **?** command to display a complete list of egress interfaces. |
| **null 0** | Optional. Routes traffic destined for the specified network to the null interface. The router drops all packets destined for the null interface. Use the null interface to allow the router to advertise a route but not forward traffic to the route. |
| <*administrative distance*> | Optional. Specifies an administrative distance associated with a particular router used to determine the best route when multiple routes to the same destination exist. The smaller the administrative distance the more preferable the route. Range is 1 to 255. |
| **track** <*name*> | Optional. Enables tracking on the indicated route. Once the named track enters a fail state, the route specified by the command is disabled and traffic will no longer be routed using that route. For more information on configuring tracks, see *track* <*name*> on page 691. |

## Default Values

By default, there are no configured routes in the route table.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 9.1 | Tunnel added as a supported interface. |
| Release 11.1 | Demand added as a supported interface. |
| Release 13.1 | Command expanded to include track feature. |

## Usage Examples

The following example adds a static route to the **10.220.0.0/16** network through the next-hop router **192.22.45.254** and a default route to **175.44.2.10**:

(config)**#ip route 10.220.0.0 255.255.0.0 192.22.45.254**
(config)**#ip route 0.0.0.0 0.0.0.0 175.44.2.10**

# ip routing

Use the **ip routing** command to enable the AOS IP routing functionality. Use the **no** form of this command to disable IP routing.

## Syntax Description

No subcommands.

## Default Values

By default, IP routing is enabled.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example enables the AOS IP routing functionality:

(config)**#ip routing**

# ip rtp firewall-traversal

Use the **ip rtp firewall-traversal** command to enable dynamic firewall traversal capability for RTP-based traffic, allowing deep packet inspection of SDP packets to occur so RTP will correctly traverse NAT in the firewall. This will open the proper ports dynamically for the RTP traffic. Use the **no** form of this command to return to the default setting. Variations of this command include:

**ip rtp firewall-traversal**
**ip rtp firewall-traversal policy-timeout** <*value*>

## Syntax Description

| | |
|---|---|
| **policy-timeout** <*value*> | Optional. Specifies timeout period in seconds allowed for inactive RTP sessions to remain in the firewall. Range is 1 to 4,294,967,295 seconds. |

## Default Values

By default, the RTP dynamic firewall traversal is disabled and the policy timeout period is 45 seconds.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Functional Notes

SIP uses the Session Description Protocol (SDP) to format the SIP message body in order to negotiate a Real-time Transport Protocol (RTP)/Real-time Transport Control Protocol (RTCP) connection between two or more User Agents (UAs). The ports used for this will always be selected in a pair, with the even port used for RTP and the odd port for RTCP.

The SIP ALG (enabled using the **ip firewall alg sip**) configures the firewall to examine the ALL SIP packets it identifies and maintain knowledge of SIP transmissions on the network. Since SIP packet headers include port information for the call setup, the ALG must intelligently read the packets and remember the information.

For a full SIP implementation, dynamic firewall traversal for RTP traffic must also be enabled using the **ip rtp firewall-traversal** command. This allows the firewall to open the proper ports for the RTP traffic between UAs. For more details on SIP functionality in the AOS, refer to the *Functional Notes* and *Technology Review* sections of the command *ip firewall alg* on page 519.

## Usage Examples

The following example enables dynamic firewall traversal and sets the policy timeout period at **60** seconds:

(config)**#ip rtp firewall-traversal policy-timeout 60**

# ip rtp qos dscp *<value>*

Use the **ip rtp qos dscp** command to configure the differentiated services code-point (DSCP) value with which to mark IP Real-time Transport Protocol (RTP) packets. This marking can then be used by the Quality of Service (QoS) mechanisms to give priority for this type of traffic in the unit. Use the **no** form of this command to disable this feature.

## Syntax Description

| | |
|---|---|
| *<value>* | Specifies the DSCP value. Valid range is 0 to 63. |

## Default Values

No default value is necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example sets the DSCP value to 63:

(config)**#ip rtp qos dscp 63**

# ip rtp udp *<number>*

Use the **ip rtp udp** command to configure a global starting User Datagram Protocol (UDP) port for Rapid Transport Protocol (RTP). Use the **no** form of this command to remove a configured UDP port.

## Syntax Description

| | |
|---|---|
| *<number>* | Specifies the value of the starting UDP port. Valid range is 1026 to 60,000. |

## Default Values

No default values.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example configures **2000** as the starting value of the UDP port:

(config)**#ip rtp udp 2000**

# ip scp server

Use the **ip scp server** command to enable the secure copy server functionality in AOS. Enabling the secure copy server allows AOS to support the transfer of files using a secure connection. A secure connection helps provide protection against outside forces gaining access to configuration files. An external secure copy server is required to facilitate the transfers from the terminal. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, the secure copy server is disabled.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example enables the secure copy server function:

(config)**#ip scp server**

# ip sdp grammar hold

Use the **ip sdp grammar hold** command to specify how to format hold messages in Session Description Protocol (SDP) announcements. Use the **no** form of this command to return to the default value.

**ip sdp grammar hold rfc2543**
**ip sdp grammar hold rfc3264**

## Syntax Description

| | |
|---|---|
| **rfc2543** | Specifies to use RFC2543 for formatting hold messages. |
| **rfc3264** | Specifies to use RFC3264 for formatting hold messages. |

## Default Values

By default, RFC2543 is used for formatting hold messages.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example specifies to use RFC3264 to format hold messages:

(config)**#ip sdp grammar hold rfc3264**

# ip sip

Use the **ip sip** command to enable the AOS Session Initiation Protocol (SIP) stack. When the SIP stack is enabled, memory is allocated for SIP functionality. For more details on SIP operation, refer to the *Technology Review* section of the command *ip firewall alg* on page 519. Use the **no** form of this command to disable the SIP stack.

## Syntax Description

No subcommands.

## Default Values

By default, the SIP stack is disabled. Refer to the *Functional Notes* section below for more details on the default state of SIP operation in AOS.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Functional Notes

By default, the AOS SIP ALG is enabled. This ALG allows the firewall to examine the ALL SIP packets it identifies and maintain knowledge of SIP transmissions on the network based on the SIP header. The SIP ALG requires the use of the SIP stack and the SIP proxy server in order to properly route SIP calls and maintain the SIP information. When the SIP ALG is enabled, the SIP stack and SIP proxy server are automatically enabled. For proper SIP operation, the firewall must also be configured to allow for dynamic holes for the RTP/RTCP traffic associated with SIP calls between User Agents (UAs). This functionality must be manually enabled using the **ip rtp firewall-traversal** command.

To completely disable SIP operation in the AOS, the following commands should be entered: **no ip firewall alg sip**, **no ip sip**, and **no ip rtp firewall-traversal**. The **no ip firewall alg sip** command disables the SIP ALG. The **no ip sip** command disables the SIP stack and frees all memory allocated to the stack.

## Usage Examples

The following example enables SIP stack:

(config)**#ip sip**

# ip sip database local

Use the **ip sip database local** command to store the location database of Session Initiation Protocol (SIP) User Agents (UAs) across a power loss using memory on the local router. Use the **no** form of the this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example specifies storing the location database on the local router:

(config)**#ip sip database local**

# ip sip location

Use the **ip sip database local** command to store the location database of Session Initiation Protocol (SIP) User Agents (UAs) across a power loss. Use the **ip sip location** command to manually add a SIP UA to the location database. Use the **no** form of the this command to disable this feature. Variations of this command include:

**ip sip location** <*username*> <*ip address*>
**ip sip location** <*username*> <*ip address*> <*number*>
**ip sip location** <*username*> <*ip address*> <*number*> <*value*>

## Syntax Description

| | |
|---|---|
| <*username*> | Specifies the user name for the UA being added to the location database. |
| <*ip address*> | Specifies the IP address for the UA being added to the location database. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| <*number*> | Optional. Specifies the UDP port of the UA to add to the database. |
| <*value*> | Optional. Specifies the time in minutes that a user is stored in the database. |

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example specifies an IP SIP location of **192.33.5.99** for a user named **2001**:

(config)**#ip sip location 2001 192.33.5.99**

# ip sip grammar alert-info url <*url*>

Use the **ip sip grammar alert-info url** command to specify the Alert-Info header host in outbound Session Initiation Protocol (SIP) messages. Use the **no** form of the command to return the setting to the default.

## Syntax Description

| | |
|---|---|
| <*url*> | Specifies an HTTP URL to be used in the Alert-Info header for IP phone tone. |

## Default Values

By default, the local loopback address is the host in the Alert-Info header (127.0.0.1).

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example sets the Alert-Info header to use a specific URL as shown in the sample header below:

(config)**#ip sip grammar alert-info url www.notused.com**

Sample header:

Alert-Info:<http://www.notused.com>;info=alert-internal

# ip sip grammar

Use the **ip sip grammar privacy** command to populate privacy lists, indicating how caller ID is handled. Use the **no** form of the command to return the setting to the default. Variations of this command include:

**ip sip grammar default-privacy critical**
**ip sip grammar default-privacy header**
**ip sip grammar default-privacy none**
**ip sip grammar default-privacy session**
**ip sip grammar default-privacy user**
**ip sip grammar restricted-privacy critical**
**ip sip grammar restricted-privacy header**
**ip sip grammar restricted-privacy none**
**ip sip grammar restricted-privacy session**
**ip sip grammar restricted-privacy user**

## Syntax Description

| | |
|---|---|
| **default-privacy** | Specifies entries into the default-privacy list for unrestricted caller ID calls. |
| **restricted-privacy** | Specifies entries into the restricted-privacy list for restricted caller ID calls. |
| **critical** | Adds critical to the Privacy header format. At least one other entry must be added to the list when using this setting. |
| **header** | Adds header to the Privacy header format. |
| **none** | Adds none to the Privacy header format. No other entries can be added to the list when using this setting. |
| **session** | Adds session to the Privacy header format. |
| **user** | Adds user to the Privacy header format. |

## Default Values

By default, both privacy lists are empty.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example sets all calls to have session privacy:

(config)**#ip sip grammar default-privacy session**

# ip sip grammar from

Use the **ip sip grammar from** command to configure the From header on Session Initiation Protocol (SIP) messages. Use the **no** form of the command to return the setting to the default. Variations of this command include:

**ip sip grammar from host domain**
**ip sip grammar from host local**
**ip sip grammar from host sip-server**
**ip sip grammar from user domestic**
**ip sip grammar from user domestic** <*Txx*>
**ip sip grammar from user international**
**ip sip grammar from user international** <*Txx*>

## Syntax Description

| | |
|---|---|
| **host** | Specifies the Host field formatting. |
| **user** | Specifies the User field formatting. |
| **domain** | Specifies the Domain for formatting the header. |
| **local** | Specifies the Local IP for formatting the header. |
| **sip-server** | Specifies the SIP server for formatting the header. |
| **domestic** | Sends the number as specified by the calling party. |
| **international** | Sends the number with E.164 formatting. |
| <*Txx*> | Optional. Indicates a two-digit trunk identifier (i.e., T01). |

## Default Values

By default, the host for formatting messages is **sip-server**. Also, the default for the user format is **domestic**.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |
| Release 13.1 | Expanded to include domestic and international formats for the From User header. |

## Functional Notes

Omitting the trunk option when issuing the **ip sip grammar from user** command specifies the User header globally.

## Usage Examples

The following example sets the From header format to use a local IP:

(config)**#ip sip grammar from host local**

The following example sets the From header format to use calling party format on trunk T02:

(config)**#ip sip grammar from user domestic T02**

## Technology Review

This technology review provides information about the E.164 recommendation for International numbering plans and telephone number formats.

A fully specified telephone number can have a maximum of 15 digits including country code, area code, and the subscriber's number. These numbers usually consist of a + prefix. E.164 numbers exclude dialing prefixes. The most familiar prefixes are international direct dialing (IDD) and national direct dialing (NDD). In countries other than the USA, the IDD and NDD are represented by different numbers.

Additionally, E.123 describes the use of + to indicate a fully specified international number. The + is used in SIP headers to provide consistency across national and international phone calls.

AOS products provide support for E.164 by being able to specify a country code and an IDD prefix. National format telephone numbers are converted to international format by prefixing them with + and the country code. On outbound international calls, + is substituted for the IDD. On incoming international calls, the + is removed. If the country code matches the configured value, it too is removed.


*Setting the From header to **international** will cause phone number to be formatted as indicated by E.164. The country code must be configured and the number must be of type **national** for this feature to work successfully.*

# ip sip grammar p-asserted-identity host

Use the **ip sip grammar p-asserted-identity host** command to enable and format the private extensions to Session Initiation Protocol (SIP) for asserted identity within trusted networks. Use the **no** form of the command to return to the default. Variations of this command include:

**ip sip grammar p-asserted-identity host domain**
**ip sip grammar p-asserted-identity host local**
**ip sip grammar p-asserted-identity host sip-server**

## Syntax Description

| | |
|---|---|
| **domain** | Specifies the domain host for formatting the header. |
| **local** | Specifies the local IP as host for formatting the header. |
| **sip-server** | Specifies the SIP server as host for formatting the header. |

## Default Values

By default, the host for formatting messages is **sip-server**.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example sets the p-asserted-identity to use a local IP:

(config)**#ip sip grammar p-asserted-identity host local**

# ip sip grammar proxy-require privacy

Use the **ip sip grammar proxy-require privacy** command to add a proxy-require header to Session Initiation Protocol (SIP) message packets containing a privacy header. Use the **no** form of the command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, this feature is disabled.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example allows a proxy-require header to be added to packets containing a privacy header:

(config)**#ip sip grammar proxy-require privacy**

# ip sip grammar request-uri

Use the **ip sip grammar request-uri** command to format the Request URI for IP SIP messages. Use the **no** form of the command to return the setting to the default. Variations of this command include:

**ip sip grammar request-uri host domain**
**ip sip grammar request-uri host sip-server**
**ip sip grammar request-uri host-resolve**

## Syntax Description

| | |
|---|---|
| **domain** | Specifies the domain for formatting the header. |
| **sip-server** | Specifies the SIP server IP for formatting the header. |
| **host-resolve** | Enables the local unit to resolve the domain before resolving the request URI. |

## Default Values

By default, the host for formatting messages is the SIP server. Also by default, **host-resolve** is disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example enables IP SIP messages to resolve the request URI from the host domain:

(config)**#ip sip grammar request-uri host domain**

The following example enables IP SIP messages to resolve the request URI from the local unit:

(config)**#ip sip grammar request-uri host-resolve**

# ip sip grammar to host

Use the **ip sip grammar to host** command to format the host format of the To header of a SIP message. Use the **no** form of the command to return the setting to the default. Variations of this command include:

**ip sip grammar to host domain**
**ip sip grammar to host sip-server**

## Syntax Description

| | |
|---|---|
| **domain** | Specifies the domain for formatting the header. |
| **sip-server** | Specifies the SIP server for formatting the header. |

## Default Values

By default, the host for formatting messages is the SIP server.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example sets the To header format to use a domain host:

(config)**#ip sip grammar to host domain**

# ip sip privacy

Use the **ip sip privacy** command to specify outbound calls to include privacy headers (when configured) and inbound calls to be filtered on privacy settings. Use the **no** form of the command to return the setting to the default.

## Syntax Description

No subcommands.

## Default Values

By default, IP SIP privacy is disabled.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example enables IP SIP privacy:

(config)**#ip sip privacy**

# ip sip qos dscp <*value*>

Use the **ip sip qos dscp** command to configure the differentiated services code-point (DSCP) value to mark IP SIP packets with. This marking can then be used by the Quality of Service (QoS) mechanisms to give priority for this type of traffic in the unit. Use the **no** form of this command to disable this feature.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the DSCP value. Valid range is 0 to 63. |

## Default Values

No default value is necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example sets the DSCP value to 63:

(config)**#ip sip qos dscp 63**

# ip sip registrar

Use the **ip sip registrar** command to configure the SIP registrar server used for registering User Agents (UAs) into the location database. For more details on SIP operation, refer to the *Technology Review* section of the command *ip firewall alg* on page 519. Use the **no** form of the **ip sip registrar** command to disable the registrar server. Variations of this command include:

**ip sip registrar**
**ip sip registrar authenticate**
**ip sip registrar default-expires** <*value*>
**ip sip registrar max-expires** <*value*>
**ip sip registrar min-expires** <*value*>
**ip sip registrar realm** <*string*>

## Syntax Description

| | |
|---|---|
| **authenticate** | Specifies that authentication is required for each UA during registration. |
| **default-expires** <*value*> | Specifies the default expiration period for the UA listing in the location database. UAs requesting registration without specifying an expiration period are given this default expiration period. Range is 0 to 2,592,000 seconds. |
| **max-expires** <*value*> | Specifies the maximum expiration period for the UA listing in the location database. All UAs registering with the SIP proxy server request an expiration period for the listing in the database. UAs requesting an expiration period between the **max-expires** and **min-expires** values are honored. Range is 0 to 2,592,000 seconds. |
| **min-expires** <*value*> | Specifies the minimum expiration period for the UA listing in the location database. All UAs registering with the SIP proxy server request an expiration period for the listing in the database. UAs requesting an expiration period between the **max-expires** and **min-expires** values are honored. Range is 0 to 2,592,000 seconds. |
| **realm** <*string*> | Specifies a realm (using an ASCII character string) for the UA listing in the location database. |

## Default Values

By default, the registrar server is disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example sets the default expiration to **5** seconds:

(config)**#ip sip registrar default-expires 5**

The following example sets the realm string:

(config)**#ip sip registrar realm voice.adtran.com**

# ip sip timer

Use the **ip sip timer** command to configure the Session Initiation Protocol (SIP) internal T1 and T2 timer (in milliseconds). Use the **no** form of this command to return to the default value. Variations of this command include:

**ip sip timer T1** <*value*>
**ip sip timer T2** <*value*>

## Syntax Description

| | |
|---|---|
| **T1** | Specifies the T1 timer. |
| **T2** | Specifies the T2 timer |
| <*value*> | Specifies time in milliseconds. Range for T1 is 50 to 1,000 milliseconds and for T2 is 1,000 to 32,000 milliseconds. |

## Default Values

By default, the T1 timer is set to 500 milliseconds and the T2 timer is set to 4,000 milliseconds.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Functional Notes

T1 is an estimate of network round trip time (RTT) and is used as the initial Invite message retransmit interval. Several SIP internal timers are derived from T1.

T2 is the maximum retransmit interval for Non-Invite requests and Invite responses.

## Usage Examples

The following example configures the T1 timer to 1,000 milliseconds:

(config)**#ip sip timer T1 1000**

# ip sip timer registration-failure-retry <*value*>

Use the **ip sip timer registration-failure-retry** command to configure the time (in seconds) that will elapse before a SIP endpoint will retry registration with the SIP server after a registration failure has occurred. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies time in seconds. Range is 10 to 604,800 seconds. |

## Default Values

By default, the registration-failure-retry timer is set to 60 seconds.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example allows a retry attempt to begin after 32 seconds:

(config)**#ip sip timer registration-failure-retry 32**

# ip sip timer rollover <*value*>

Use the **ip sip timer rollover** command to set the SIP timer for Invite transactions. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies time in seconds. Range is 1 to 32 seconds. |

## Default Values

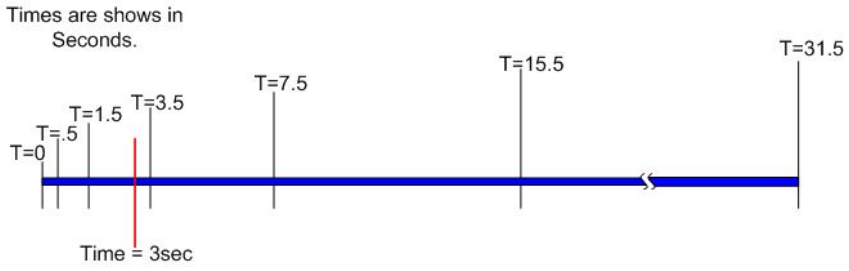
By default, the rollover timer is set to 3 seconds.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Functional Notes

The **ip sip timer rollover** command sets the SIP timer B value for Invite transactions originating from a SIP trunk. When originating a call, the SIP trunk attempts to send Invite messages to the primary SIP server and waits for a response. If there is no response, the SIP trunk waits for 0.5 seconds before attempting to send another Invite to the same SIP server. If no response, the SIP trunk waits for 1 second before attempting to send another Invite, then waits 2 seconds, and so on. These increasing intervals are shown in diagram below.




The rollover timer allows the user to control how long to wait before trying the next server. In the diagram above, the red line indicates the rollover timer expiration. If there is no response after the timer expires, the SIP trunk will attempt to send Invite messages to the highest priority backup SIP server obtained via DNS SRV. The SIP trunk starts over at T=0 with the next server and doesn't send any more messages to the timed out server. As long the SIP trunk does not receive a response, it will continue this cycle until it has attempted to contact all the SIP servers.

## Usage Examples

The following example allows connection attempts to continue for up to 32 seconds before rolling over to the backup server:

(config)**#ip sip timer rollover 32**

# ip snmp agent

Use the **ip snmp agent** command to enable the Simple Network Management Protocol (SNMP) agent. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, the SNMP agent is disabled.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

Allows a MIB browser to access standard MIBs within the product. This also allows the product to send traps to a trap management station.

## Usage Examples

The following example enables the IP SNMP agent:

(config)**#ip snmp agent**

# ip sntp server

Use the **ip sntp server** command to enable the simple network time protocol (SNTP) server. This allows the unit to accept SNTP requests. Use the **no** form of this command to disable the server.

## Syntax Description

No subcommands.

## Default Values

By default, the SNTP server is disabled.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example enables the SNTP server:

(config)**#ip sntp server**

# ip sntp source-interface *<interface>*

Use the **ip sntp source-interface** command to specify a source interface for SNTP traffic originated by the unit. The IP address of the specified interface will be used to source all SNTP traffic. Use the **no** form of this command if you do not wish to override the default source IP address.

## Syntax Description

| | |
|---|---|
| *<interface>* | Specifies the source interface for SNTP traffic. Specify an interface in the format *<interface type [slot/port \| slot/port.sub-interface id \| interface id \| interface id.sub-interface id]>*. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **ip sntp source-interface ?** for a complete list of valid interfaces. |

## Default Values

No default value is necessary for this command.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Functional Notes

This command allows you to override the *sender* field in the IP packet. If you have multiple interfaces in your unit, changing the *sender* tells the receiver where to send replies. This functionality can also be used to allow packets to get through firewalls that would normally block the flow.

## Usage Examples

The following example configures the unit to use the **loopback 1** interface as the source IP for SNTP traffic:

(config)**#ip sntp source-interface loopback 1**

# ip subnet-zero

The **ip subnet-zero** command is the default operation and cannot be disabled. This command signifies the router's ability to route to subnet-zero subnets.

## Syntax Description

No subcommands.

## Default Values

By default, this command is enabled.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Usage Examples

The following example **subnet-zero** is enabled:

(config)**#ip subnet-zero**

# ip tacacs source-interface *<interface>*

Use the **ip tacacs source-interface** command to specify a source interface for TACACS+ traffic originated by the unit. The IP address of the specified interface will be used to source all TACACS+ traffic. Use the **no** form of this command if you do not wish to override the default source IP address.

## Syntax Description

| | |
|---|---|
| *<interface>* | Specifies the source interface for TACACS+ traffic. Specify an interface in the format *<interface type [slot/port \| slot/port.sub-interface id \| interface id \| interface id.sub-interface id]>*. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **ip tacacs source-interface ?** for a complete list of valid interfaces. |

## Default Values

No default value is necessary for this command.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Functional Notes

This command allows you to override the *sender* field in the IP packet. If you have multiple interfaces in your unit, changing the *sender* tells the receiver where to send replies. This functionality can also be used to allow packets to get through firewalls that would normally block the flow.

## Usage Examples

The following example configures the unit to use the **loopback 1** interface as the source IP for TACACS+ traffic:

(config)**#ip tacacs source-interface loopback 1**

# ip tftp server

Use the **ip tftp server** to enable the Trivial File Transfer Protocol (TFTP) server. Use the **no** form of this command to disable the TFTP server. Variations of this command include:

**ip tftp server**
**ip tftp server access-class** <*name*> **in**
**ip tftp server overwrite**

## Syntax Description

| | |
|---|---|
| **access-class** <*name*> **in** | Controls access to the internal TFTP server using the specified access control list. |
| **overwrite** | Enables the TFTP server to overwrite existing files. |

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |
| Release 13.1 | Command was expanded to include the overwrite feature. |

## Usage Examples

The following example enables the TFTP server:

(config)**#ip tftp server**

The following example enables the TFTP server as well as self-bound TFTP incoming connections. In this example, **classname** is the name of the access class.

(config)**#ip tftp server access-class classname in**

# ip ftp server default-filesystem

Use the **ip ftp server default-filesystem** to enable the File Transfer Protocol (FTP) server. Use the **no** form of this command to disable the FTP server.

## Syntax Description

| | |
|---|---|
| **cflash** | Specifies FTP server to use the compact flash as the default file system. |
| **flash** | Specifies FTP server to use the flash as the default file system. |

## Default Values

By default, this command is set to **ip ftp server default-filesystem flash**.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example enables the default FTP server **cflash** file system:

(config)**#ip ftp server default-filesystem cflash**

# ip tftp source-interface *<interface>*

Use the **ip tftp source-interface** command to use the specified interface's IP address as the source IP address for Trivial File Transfer Protocol (TFTP) traffic transmitted by the unit. Use the **no** form of this command if you do not wish to override the normal source IP address.

## Syntax Description

| | |
|---|---|
| *<interface>* | Specifies the interface to be used as the source IP address for TFTP traffic. Specify an interface in the format *<interface type [slot/port \| slot/port.sub-interface id \| interface id \| interface id.sub-interface id]>*. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **ip tftp source-interface ?** for a complete list of valid interfaces. |

## Default Values

No default value is necessary for this command.

## Default Values

No default value is necessary for this command.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Functional Notes

This command allows you to override the *sender* field in the IP packet. If you have multiple interfaces in your unit, changing the *sender* tells the receiver where to send replies. This functionality can also be used to allow packets to get through firewalls that would normally block the flow.

## Usage Examples

The following example configures the unit to use the **loopback 1** interface as the source IP for TFTP traffic:

(config)**#ip tftp source-interface loopback 1**

# ip urlfilter <*name*> http

Use the **ip urlfilter http** command to create a URL filter for Hypertext Transfer Protocol (HTTP) (TCP port 80) traffic. Use the **no** form of this command to delete the specified HTTP URL filter.

*The URL filtering software runs on a server independent of the AOS product. For additional information about the URL filtering technology, refer to the vendor's website.*

## Syntax Description

| | |
|---|---|
| <*name*> | Specifies the URL filter name. |

## Default Values

By default, no URL filters are configured.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Functional Notes

The firewall must be enabled using the **ip firewall** command in order to use URL filters. The URL filter must be applied to the appropriate interface by using the **ip urlfilter** <*filtername*> **[in | out]** command. Refer to this command in the appropriate interface for more information.

## Usage Examples

The following example creates the HTTP URL filter called **MyFilter** that can be applied to an interface for content filtering:

(config)**#ip urlfilter MyFilter http**

# ip urlfilter allowmode

Use the **ip urlfilter allowmode** command to allow all URL requests in cases when all URL filter servers are down. Use the **no** form of this command to block all URL requests when all URL filter servers are down.

## Syntax Description

No subcommands.

## Default Values

By default, all URL requests will be blocked when all URL filter servers are down.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Example

The following example permits all URL requests even when URL filter servers are down:

(config)**#ip urlfilter allowmode**

# ip urlfilter exclusive-domain

Use the **ip urlfilter exclusive-domain** command to instruct AOS to always allow or always block a domain without first having to verify with the URL filter server. Use the **no** form of this command to remove an exclusive domain. Variations of this command include:

**ip urlfilter exclusive-domain deny** *<name>*
**ip urlfilter exclusive-domain permit** *<name>*

## Syntax Description

| | |
|---|---|
| **deny** *<name>* | Specifies that the domain name be blocked without verifying with the URL filter server. |
| **permit** *<name>* | Specifies that the domain name be allowed without verifying with the URL filter server. |

## Default Values

By default, no exclusive domains are configured.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Functional Notes

Domain matching is based on an exact match between the HTTP header and entries in the **ip urlfilter exclusive-domain** command. In order to exactly match requests destined for a domain, entries should list all possible variations of the domain that would appear in the **Host** field of an HTTP header. Refer to the *Usage Examples* section of this command for more detailed information.

## Usage Example

The following example will always allow access to **www.adtran.com** and **adtran.com** without first having to verify the domain with the URL filter server:

(config)**#ip urlfilter exclusive-domain permit www.adtran.com**
(config)**#ip urlfilter exclusive-domain permit adtran.com**

The following example will always block access to **www.localnews.com** without first having to verify the domain with the URL filter server:

(config)**#ip urlfilter exclusive-domain deny www.localnews.com**

# ip urlfilter max-request <*value*>

Use the **ip urlfilter max-request** command to set the maximum number of outstanding URL lookup requests that can be sent to a URL filter server without a response. Use the **no** form of this command to set the value back to its default.

## Syntax Description

| | |
|---|---|
| <*value*> | The maximum number of outstanding URL lookup requests. Valid range is 1 to 500 requests. |

## Default Values

By default, the number of outstanding requests is 500.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Functional Notes

After the maximum number of URL lookup requests is reached, the **no ip urlfilter allowmode** setting will be used to allow or block all following requests until enough URL lookup responses have been received from the URL filter server.

## Usage Example

The following example sets the maximum number of URL lookup requests to **250**:

(config)**#ip urlfilter max-request 250**

# ip urlfilter max-response *<value>*

Use the **ip urlfilter max-response** command to set the maximum number of responses allowed to buffer before receiving an allow or block status from the URL filter server. Use the **no** form of this command to set the value back to its default.

## Syntax Description

| | |
|---|---|
| *<value>* | Specifies the maximum number of responses allowed to buffer. Valid range is 1 to 100 responses. |

## Default Values

By default, the value of buffered responses is 100.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Functional Notes

When a URL request comes through the unit and URL filtering is enabled, a lookup request is sent to the URL filter server and the HTTP request is forwarded to the HTTP server at the same time. If the HTTP server responds before the URL filter server, the response must be buffered until the URL filter server responds with **allow** or **block**. Once the maximum number of buffered HTTP responses is reached, all following HTTP responses are dropped until some of the existing buffered responses are released. Buffered responses are released when the URL filter server sends a response or when the firewall association times out.

## Usage Example

The following example sets the maximum number of buffered responses to **250**:

(config)**#ip urlfilter max-response 250**

# ip urlfilter server <*ip address*>

Use the **ip urlfilter server** command to identify a URL filter server by IP address and port number. Use the **no** form of this command to remove the server from use. Variations of this command include:

**ip urlfilter server** <*ip address*>
**ip urlfilter server** <*ip address*> **port** <*number*>
**ip urlfilter server** <*ip address*> **timeout** <*value*>

## Syntax Description

| | |
|---|---|
| <*ip address*> | Specifies the server IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| **port** <*number*> | Specifies the server TCP port number which will receive requests. |
| **timeout** <*value*> | Specifies the number of seconds to wait for a response from the URL filtering server before determining that it is out of service. Range is 1 to 300 seconds. |

## Default Values

By default, there are no URL filtering servers configured. When configuring a URL filtering server, the port default is 15,868, and the timeout default is 5 seconds.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Example

The following example identifies a URL filtering server at IP address **10.1.1.1** that listens for URL filtering requests on port 15,868 (default) and waits for a response for **10** seconds before determining that the filtering server is down:

(config)**#ip urlfilter server 10.1.1.1 timeout 10**

# isdn-group

Use the **isdn-group** command to enter the ISDN Group Configuration mode command set. Use the **no** form of this command to disable this feature. Refer to the section *ISDN Group Configuration Command Set* on page 1735 for more information on the commands available for each group.

## Syntax Description

No subcommands.

## Default Values

No default values necessary for this command.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Functional Notes

An ISDN group allows the user to specify the maximum and minimum number of B-channels that can be used for a specific type of call. It is a logical group of B-channels from one or more ISDN interfaces. The interfaces can be of different types (e.g., PRI and BRI). An ISDN interface can be a member of multiple ISDN groups which makes it possible to share its B-channels between different types of calls.

## Usage Examples

The following example uses the **isdn-group** command to enter the ISDN Group Configuration mode:

```
(config)#isdn-group
(config-isdn-group 1)#
```

# isdn-number-template

Use the **isdn-number-template** command to create an entry in the ISDN number type template that is used when encoding the called party and calling party information elements for inbound and outbound ISDN calls. Use the **no** form of the command to delete the configured entry. Variations of this command include the following:

**isdn-number-template** <*template id*> **prefix** <*number*> **abbreviated** <*pattern*>
**isdn-number-template** <*template id*> **prefix** <*number*> **international** <*pattern*>
**isdn-number-template** <*template id*> **prefix** <*number*> **national** <*pattern*>
**isdn-number-template** <*template id*> **prefix** <*number*> **network-specific** <*pattern*>
**isdn-number-template** <*template id*> **prefix** <*number*> **subscriber** <*pattern*>
**isdn-number-template** <*template id*> **prefix** <*number*> **unknown** <*pattern*>

## Syntax Description

| | |
|---|---|
| <*template id*> | Specifies a numeric identifier for the template entry. Valid range is 1 to 255. |
| **prefix** <*number*> | Specifies the expected prefix for the call type. Prefixes can be left blank (using double quotation marks “”) or consist of unlimited length strings of 0s and 1s. For example, for international calls made from within the United States, a prefix of 011 is expected. |
| **abbreviated** | Specifies to use Abbreviated in the Type of Number octet (bits 110). Abbreviated is used mainly in private ISDN network applications and the implementation is network dependent. |
| **international** | Specifies to use International in the Type of Number octet (bits 001). International is used for calls destined outside the national calling area. International calls have the international direct dialing prefix removed. For example, consider an international call of 011-N$, where the international direct dialing prefix is 011 and the N$ represents the digits necessary for routing the call at the destination. When the Called Party IE is created for this call, the prefix is stripped and the N$ digits are placed in the Number Digits field. |
| **national** | Specifies to use National (bits 010) in the Type of Number octet. National is used for calls destined for inside the national calling area (i.e., does not cross into an international LATA). National calls have the direct dialing prefix removed. For example, consider a national call with a direct dialing prefix of 1 and NXX-NXX-XXXX to represent the ten-digit number necessary for routing the call. When the Called Party IE is created for this call, the prefix (1) is stripped and the NXX-NXX-XXXX digits are placed in the Number Digits field. |

**network-specific** Specifies to use Network-Specific (bits 011) in the Type of Number octet. Network-Specific is used for calls that require special access to a private network which requires the use of a prefix that should be stripped once access to the network has been gained. Network-Specific calls have the dialing prefix removed. For example, a call to a private network with the 700 consists of 700-N$, where 700 is the dialing prefix and N$ represents the digits necessary for routing the call at the destination. When the Called Party IE is created for this call, the prefix is stripped and the N$ is placed in the Number Digits field.

**subscriber** Specifies to use Subscriber (bits 100) in the Type of Number octet. Subscriber is used for local calls (not long-distance). Subscriber calls, by default, have the area code removed. For example, a subscriber call to 916-555-1212 would have the prefix 916 stripped and 555-1212 in the Number Digits field. For areas with mandatory ten-digit dialing, a blank prefix should be entered to ensure that all ten digits are passed to the Number Digits field.

**unknown** Specifies to use Unknown (bits 000) in the Type of Number octet. Unknown is used when the number type is not known. Unknown numbers are assumed to have no prefix, and the entire dialed number is presented in the Number Digits field.

*<pattern>* Specifies a pattern for this template.

Valid Characters:

| | |
|---|---|
| 0-9 | Match exact digit only. |
| X | Match any single digit 0 through 9. |
| N | Match any single digit 2 through 9. |
| M | Match any single digit 1 through 8. |
| [ ] | Match any digit in the list. For example: [1,4,6] matches 1, 4, and 6 only. [1-3, 5] matches 1, 2, 3, and 5. |

## Default Values

The following default number template entry exists for domestic emergency calls (911):

**isdn-number-template 0 prefix " " subscriber 911**

## Functional Notes

The following is an example number type template:

| Prefix | Pattern | Type |
|---|---|---|
| " " | NXX-XXXX | Subscriber |
| "1" | NXX-NXX-XXXX | National |
| "011" | X$ | International |
| " " | N11 | Subscriber (i.e. 411, 911, etc.) |

## Usage Examples

The following example creates a number template (labeled **1**) and prefix (labeled **1**) for national calls:

(config)**#isdn-number-template 1 prefix 1 national Nxx-Nxx-xxxx**

# line

Use the **line** command to enter the line configuration for the specified console, Telnet, or secure shell (SSH) session. Refer to the sections *Line (Console) Interface Config Command Set* on page 732, *Line (Telnet) Interface Config Command Set* on page 759, and *Line (SSH) Interface Config Command Set* on page 748 for information on the subcommands. Variations of this command include:

**line console** *<line number>*
**line ssh** *<line number>*
**line ssh** *<line number> <ending number>*
**line telnet** *<line number>*
**line telnet** *<line number> <ending number>*

## Syntax Description

| | |
|---|---|
| **console** | Enters the configuration mode for the DB-9 (female) **CONSOLE** port located on the rear panel of the unit. Refer to the section *Line (Console) Interface Config Command Set* on page 732 for information on the subcommands found in that command set. |
| **telnet** | Enters the configuration mode for Telnet session(s), allowing you to configure for remote access. Refer to the section *Line (Telnet) Interface Config Command Set* on page 759 for information on the subcommands found in that command set. |
| **ssh** | Enters the configuration mode for SSH. Refer to the section *Line (SSH) Interface Config Command Set* on page 748 for information on the subcommands found in that command set. |
| *<line number>* | Specifies the starting session to configure for remote access. Valid range for console is 0. Valid range for Telnet and SSH is 0 to 4.<br>If configuring a single Telnet or SSH session, enter a single line number. |
| *<ending number>* | Optional. Specifies the last Telnet or SSH session to configure for remote access. Valid range is 0 to 4.<br>For example, to configure all available Telnet sessions, enter **line telnet 0 4**. |

## Default Values

By default, there are no configured Telnet or SSH sessions. By default, the AOS line console parameters are configured as follows:

Data Rate: 9600
Data bits: 8
Stop bits: 1
Parity Bits: 0
No flow control

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 11.1 | Command was expanded to include SSH. |

## Usage Examples

The following example begins the configuration for the **CONSOLE** port located on the rear of the unit:

```
(config)#line console 0
(config-con0)#
```

The following example begins the configuration for all available Telnet sessions:

```
(config)#line telnet 0 4
(config-telnet0-4)#
```

The following example begins the configuration for all available SSH sessions:

```
(config)#line ssh 0 4
(config-ssh0-4)#
```

# lldp

Use the **lldp** command to configure global settings that control the way LLDP functions. Use the **no** form of this command to return to the default settings. Variations of this command include:

**lldp minimum-transmit-interval** <*value*>
**lldp reinitialization-delay** <*value*>
**lldp transmit-interval** <*value*>
**lldp ttl-multiplier** <*value*>

## Syntax Description

| | |
|---|---|
| **minimum-transmit-interval** | Defines the minimum amount of time between transmission of LLDP frames in seconds. Range is 1 to 8192 seconds. |
| **reinitialization-delay** | Defines the minimum amount of time to delay after LLDP is disabled on a port before allowing transmission of additional LLDP frames on that port in seconds. Range is 1 to 10 seconds. |
| **transmit-interval** | Defines the delay between LLDP frame transmission attempts during normal operation in seconds. Range is 5 to 32,768 seconds. |
| **ttl-multiplier** | Defines the multiplier to be applied to the transmit interval to compute the time-to-live for data sent in an LLDP frame. Range is 2 to 10. |
| <*value*> | Specifies the interval, delay, or multiplier. |

## Default Values

By default, **minimum-transmit-interval** is 2 seconds; **reinitialization-delay** is 2 seconds; **transmit-interval** is 30 seconds; and **ttl-multiplier** is 4.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Functional Notes

Once a device receives data from a neighboring device in an LLDP frame, it will retain that data for a limited amount of time. This amount of time is called time-to-live, and it is part of the data in the LLDP frame. The time-to-live transmitted in the LLDP frame is equal to the transmit interval multiplied by the TTL multiplier.

## Usage Examples

The following example sets the LLDP minimum transmit interval to 10 seconds:

(config)**#lldp minimum-transmit-interval 10**

The following example sets the LLDP reinitialization delay to 5 seconds:

(config)**#lldp reinitialization-delay 5**

The following example sets the LLDP transmit interval to 15 seconds:

(config)**#lldp transmit-interval 15**

The following example sets the LLDP TTL multiplier to 2 and the time-to-live for all LLDP frames transmitted from this unit to 30 seconds;

(config)**#lldp transmit-interval 15**
(config)**#lldp ttl-multiplier 2**

# logging console

Use the **logging console** command to enable the AOS to log events to all consoles. Use the **no** form of this command to disable console event logging.

## Syntax Description

No subcommands.

## Default Values

By default, logging console is disabled.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following example enables the AOS to log events to all consoles:

(config)**#logging console**

# logging email address-list <*email address*> ; <*email address*>

Use the **logging email address-list** command to specify one or more email addresses that will receive notification when an event matching the criteria configured using the **logging email priority-level** command is logged by the AOS. Refer to *logging email priority-level* on page 616 for more information. Use the **no** form of this command to remove a listed address.

## Syntax Description

| | |
|---|---|
| <*email address*> | Specifies the complete email address to use when sending logged messages. (This field allows up to 256 characters.)<br>Enter as many email addresses as desired, placing a semi-colon (;) between addresses. |

## Default Values

By default, there are no configured logging email addresses.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example specifies three email addresses to use when sending logged messages:

(config)**#logging email address-list**
**admin@adtranemail.com;ntwk@adtranemail.com;support@adtranemail.com**

# logging email exception-report address-list *<email address>* ; *<email address>*

Use the **logging email exception-report address-list** command to specify one or more email addresses to receive an exception report for use in troubleshooting. Use the **no** form of this command to remove a listed address.

## Syntax Description

| | |
|---|---|
| *<email address>* | Specifies the complete email address to use when sending exception reports. (This field allows up to 256 characters.)<br>Enter as many email addresses as desired, placing a semi-colon (;) between addresses. |

## Default Values

By default, there are no configured logging email addresses.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Functional Notes

When AOS experiences an exception it will generate a file with detailed information that ADTRAN's Technical Support can use to diagnose the problem, This command allows the unit to email the exception report to a list of addresses upon rebooting after the exception. This command should be used in conjunction with the other logging email commands. Refer to *logging email address-list <email address> ; <email address>* on page 613, *logging email on* on page 615, *logging email priority-level* on page 616, *logging email receiver-ip <ip address>* on page 617, *logging email sender* on page 618, and *logging email source-interface <interface>* on page 619 for more information.

## Usage Examples

The following example will enable exception report forwarding to **john.doe@company.com** using the **1.1.1.1** SMTP email server:

(config)**#logging email on**
(config)**#logging email receiver-ip 1.1.1.1**
(config)**#logging email exception-report address-list john.doe@company.com**

# logging email on

Use the **logging email on** command to enable the AOS email event notification feature. Use the **logging email address-list** command to specify email address(es) that will receive notification when an event matching the criteria configured using the **logging email priority-level** command is logged by the AOS. Refer to *logging email priority-level* on page 616 for more information. Use the **no** form of this command to disable the email notification feature.

## Syntax Description

No subcommands.

## Default Values

By default, email event notification is disabled.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

The domain name is appended to the sender name when sending event notifications. Refer to the command *ip domain-name <name>* on page 509 for related information.

## Usage Examples

The following example enables the AOS email event notification feature:

(config)**#logging email on**

# logging email priority-level

Use the **logging email priority-level** command to set the threshold for events sent to the addresses specified using the **logging email address-list** command. All events with the specified priority or higher will be sent to all addresses in the list. The **logging email on** command must be enabled. Refer to *logging email address-list <email address> ; <email address>* on page 613 and *logging email on* on page 615 for related information. Use the **no** form of this command to return to the default priority. Variations of this command include:

**logging email priority-level error**
**logging email priority-level fatal**
**logging email priority-level info**
**logging email priority-level notice**
**logging email priority-level warning**

## Syntax Description

Sets the minimum priority threshold for sending messages to email addresses specified using the **logging email address-list** command. The following priorities are available (ranking from lowest to highest):

| | |
|---|---|
| **error** | Logs events with **error** and **fatal** priorities. |
| **fatal** | Logs only events with a **fatal** priority. |
| **info** | Logs all events. |
| **notice** | Logs events with **notice**, **warning**, **error**, and **fatal** priorities. |
| **warning** | Logs events with **warning**, **error**, and **fatal** priorities. |

## Default Values

By default, the **logging email priority-level** is set to **warning**.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example sends all messages with **warning** level or greater to the email addresses listed using the **logging email address-list** command:

(config)**#logging email priority-level warning**

# logging email receiver-ip *<ip address>*

Use the **logging email receiver-ip** command to specify the IP address of the email server to use when sending notification that an event matched the criteria configured using the **logging email priority-level** command. Refer to *logging email priority-level* on page 616 for related information. Use the **no** form of this command to remove a configured address.

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies the IP address of the mail server to use when sending logged messages. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

## Default Values

By default, there are no configured email server addresses.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example specifies an email server (with address 172.5.67.99) to use when sending logged messages:

(config)**#logging email receiver-ip 172.5.67.99**

# logging email sender

Use the **logging email sender** command to specify the sender in an outgoing email message. This name will appear in the **From** field of the receiver's inbox. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

No default value is necessary for this command.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Usage Examples

The following example sets a sender for outgoing messages:

(config)**#logging email sender myUnit@myNetwork.com**

# logging email source-interface *<interface>*

Use the **logging email source-interface** command to use the specified interface's IP address as the source IP address for email messages transmitted by the unit. Use the **no** form of this command if you do not wish to override the normal source IP address.

## Syntax Description

| | |
|---|---|
| *<interface>* | Specifies the interface to be used as the source IP address for email messages. Specify an interface in the format *<interface type [slot/port \| slot/port.sub-interface id \| interface id \| interface id.sub-interface id]>*. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **logging email source-interface ?** for a complete list of valid interfaces. |

## Default Values

No default value is necessary for this command.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Functional Notes

This command allows you to override the *sender* field in the IP packet. If you have multiple interfaces in your unit, changing the *sender* tells the receiver where to send replies. This functionality can also be used to allow packets to get through firewalls that would normally block the flow.

## Usage Examples

The following example configures the unit to use the **loopback 1** interface as the source IP for email messages:

(config)**#logging email source-interface loopback 1**

# logging facility *<type>*

Use the **logging facility** command to specify a syslog facility type for the syslog server. Error messages meeting specified criteria are sent to the syslog server. For this service to be active, you must enable log forwarding. Refer to *logging forwarding on* on page 621 for related information. Facility types are described under *Functional Notes* below. Use the **no** form of this command to return it to its default setting.

## Syntax Description

| | | |
|---|---|---|
| *<type>* | Specifies the syslog facility type (refer to *Functional Notes* below). The following is a list of all the valid facility types: | |
| | **auth** | Authorization system |
| | **cron** | Cron facility |
| | **daemon** | System daemon |
| | **kern** | Kernel |
| | **local0-local7** | Reserved for locally-defined messages |
| | **lpr** | Line printer system |
| | **mail** | Mail system |
| | **news** | USENET news |
| | **sys9 - sys14** | System use |
| | **syslog** | System log |
| | **user** | User process |
| | **uucp** | UNIX-to-UNIX copy system |

## Default Values

The default value is **local7**.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Usage Examples

The following example configures the syslog facility to the cron facility type:

(config)**#logging facility cron**

# logging forwarding on

Use the **logging forwarding on** command to enable the AOS syslog event feature. Use the **logging forwarding priority-level** command to specify the event matching the criteria used by the AOS to determine whether a message should be forwarded to the syslog server. Refer to *logging forwarding priority-level* on page 622 for related information. Use the **no** form of this command to disable the syslog event feature.

## Syntax Description

No subcommands.

## Default Values

By default, syslog event notification is disabled.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example enables the AOS syslog event feature:

(config)**#logging forwarding on**

# logging forwarding priority-level

Use the **logging forwarding priority-level** command to set the threshold for events sent to the configured syslog server specified using the **logging forwarding receiver-ip** command. All events with the specified priority or higher will be sent to all configured syslog servers. Refer to *logging email priority-level* on page 616 for more information. Use the **no** form of this command to return to the default priority. Variations of this command include:

**logging forwarding priority-level error**
**logging forwarding priority-level fatal**
**logging forwarding priority-level info**
**logging forwarding priority-level notice**
**logging forwarding priority-level smdr**
**logging forwarding priority-level warning**

## Syntax Description

Sets the minimum priority threshold for sending messages to the syslog server specified using the **logging forwarding receiver-ip** command. The following priorities are available (ranking from lowest to highest):

| | |
|---|---|
| **error** | Logs events with **error** and **fatal** priorities. |
| **fatal** | Logs only events with a **fatal** priority. |
| **info** | Logs all events. |
| **notice** | Logs events with **notice**, **warning**, **error**, and **fatal** priorities. |
| **smdr** | Logs events with **smdr** priorities. |
| **warning** | Logs events with **warning**, **error**, and **fatal** priorities. |

## Default Values

By default the **logging forwarding priority-level** is set to **warning**.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 12.1 | Command was expanded to include **smdr**. |

## Usage Examples

The following example sends all messages with **warning** level or greater to the syslog server listed using the **logging forwarding receiver-ip** command:

(config)**#logging forwarding priority-level warning**

# logging forwarding receiver-ip *<ip address>*

Use this **logging forwarding receiver-ip** command to specify the IP address of the syslog server to use when logging events that match the criteria configured using the **logging forwarding priority-level** command. Enter multiple **logging forwarding receiver-ip** commands to develop a list of syslog servers to use. Refer to *logging forwarding priority-level* on page 622 for related information. Use the **no** form of this command to remove a configured address.

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies the IP address of the syslog server to use when logging messages. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

## Default Values

By default, there are no configured syslog server addresses.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example specifies a syslog server (with address **172.5.67.99**) to use when logging messages:

(config)**#logging forwarding receiver-ip 172.5.67.99**

# logging forwarding source-interface *<interface>*

Use the **logging forwarding source-interface** command to configure the specified interface's IP address as the source IP address for the syslog server to use when logging events. Use the **no** form of this command if you do not wish to override the normal source IP address.

## Syntax Description

| | |
|---|---|
| *<interface>* | Specifies the interface to be used as the source IP address for event log traffic. Specify an interface in the format *<interface type [slot/port \| slot/port.sub-interface id \| interface id \| interface id.sub-interface id]>*. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **logging forwarding source-interface?** for a complete list of valid interfaces. |

## Default Values

No default value is necessary for this command.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Functional Notes

This command allows you to override the *sender* field in the IP packet. If you have multiple interfaces in your unit, changing the *sender* tells the receiver where to send replies. This functionality can also be used to allow packets to get through firewalls that would normally block the flow.

## Usage Examples

configures the unit to use the **loopback 1** interface as the source IP for event log traffic:

(config)**#logging forwarding source-interface loopback 1**

# mac address-table aging-time *<value>*

Use the **mac address-table aging-time** command to set the length of time dynamic MAC addresses remain in the switch or bridge forwarding table. Use the **no** form of this command to reset this length to its default.

## Syntax Description

| | |
|---|---|
| *<value>* | Set an aging time in seconds. Range is 10 to 1,000,000 seconds. Set to 0 to disable the timeout. |

## Default Values

By default, the aging time is 300 seconds.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following example sets the aging time to 10 minutes:

(config)**#mac address-table aging-time 600**

# mac address-table static <*mac address*> bridge <*bridge id*> interface <*interface*>

Use the **mac address-table static bridge interface** command to insert a static MAC address entry into the bridge forwarding table. Use the **no** form of this command to remove an entry from the table.

## Syntax Description

| | |
|---|---|
| *<mac address>* | Specifies a valid 48-bit MAC address. MAC addresses should be expressed in following format xx:xx:xx:xx:xx:xx (for example, **00:A0:C8:00:00:01**). |
| *<bridge id>* | Specifies a valid bridge interface ID. |
| *<interface>* | Specifies a valid interface in the format *<interface type [slot/port \| slot/port.sub-interface id \| interface id \| interface id.sub-interface id]>*. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **mac address-table static bridge interface ?** for a complete list of valid interfaces. |

## Default Values

By default, there are no static entries configured.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example adds a static MAC address to PPP 1 on bridge 4:

(config)**#mac address-table static 00:A0:C8:00:00:01 bridge 4 interface ppp 1**

# mac address-table static <*mac address*> vlan <*vlan id*> interface <*interface*>

Use the mac **address-table static vlan interface** command to insert a static MAC address entry into the MAC address table. Use the **no** form of this command to remove an entry from the table.

## Syntax Description

| | |
|---|---|
| *<mac address>* | Specifies a valid 48-bit MAC address. MAC addresses should be expressed in following format xx:xx:xx:xx:xx:xx (for example, **00:A0:C8:00:00:01**). |
| *<vlan id>* | Specifies a valid VLAN interface ID. Range is 1 to 4094. |
| *<interface>* | Specifies a valid interface in the format *<interface type [slot/port \| slot/port.sub-interface id \| interface id \| interface id.sub-interface id]>*. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **mac address-table static vlan interface ?** for a complete list of valid interfaces. |

## Default Values

By default, there are no static entries configured.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following example adds a static MAC address to Ethernet 0/1 on VLAN 4:

(config)**#mac address-table static 00:A0:C8:00:00:01 00:12:79:00:00:01 vlan 4 interface ethernet 0/1**

# modem countrycode <*value*>

Use the **modem countrycode** command to specify the modem configuration for the applicable country.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the modem configuration for the applicable country. Refer to *Functional Notes* for countrycode values. |

## Default Values

By default, **modem countrycode** is set to **USA/Canada**.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Functional Notes

The following country codes are available for modem configuration:

| | |
|---|---|
| Algeria | - Algeria Modem configuration |
| Argentina | - Argentina Modem configuration |
| Australia | - Australia Modem configuration |
| Austria | - Austria Modem configuration |
| Bahrain | - Bahrain Modem configuration |
| Belgium | - Belgium Modem configuration |
| Bolivia | - Bolivia Modem configuration |
| Brazil | - Brazil Modem configuration |
| Chile | - Chile Modem configuration |
| China | - China Modem configuration |
| Colombia | - Colombia Modem configuration |
| Costa_Rica | - Costa_Rica Modem configuration |
| Cyprus | - Cyprus Modem configuration |
| Czechoslovakia | - Czechoslovakia Modem configuration |
| Denmark | - Denmark Modem configuration |
| Ecuador | - Ecuador Modem configuration |
| Egypt | - Egypt Modem configuration |
| Finland | - Finland Modem configuration |
| France | - France Modem configuration |
| Germany | - Germany Modem configuration |
| Greece | - Greece Modem configuration |
| Guatemala | - Guatemala Modem configuration |
| Hong_Kong | - Hong_Kong Modem configuration |
| Hungary | - Hungary Modem configuration |
| India | - India Modem configuration |
| Indonesia | - Indonesia Modem configuration |
| Ireland | - Ireland Modem configuration |

| | |
|---|---|
| Israel | - Israel Modem configuration |
| Italy | - Italy Modem configuration |
| Japan | - Japan Modem configuration |
| Jordan | - Jordan Modem configuration |
| Korea | - Korea Modem configuration |
| Kuwait | - Kuwait Modem configuration |
| Lebanon | - Lebanon Modem configuration |
| Malaysia | - Malaysia Modem configuration |
| Mexico | - Mexico Modem configuration |
| Morocco | - Morocco Modem configuration |
| Netherlands | - Netherlands Modem configuration |
| New_Zealand | - New_Zealand Modem configuration |
| Norway | - Norway Modem configuration |
| Oman | - Oman Modem configuration |
| Panama | - Panama Modem configuration |
| Peru | - Peru Modem configuration |
| Philippines | - Philippines Modem configuration |
| Poland | - Poland Modem configuration |
| Portugal | - Portugal Modem configuration |
| Puerto_Rico | - Puerto_Rico Modem configuration |
| Qatar | - Qatar Modem configuration |
| Russia | - Russia Modem configuration |
| Saudi_Arabia | - Saudi_Arabia Modem configuration |
| Singapore | - Singapore Modem configuration |
| Slovakia | - Slovakia Modem configuration |
| Slovenia | - Slovenia Modem configuration |
| South_Africa | - South_Africa Modem configuration |
| Spain | - Spain Modem configuration |
| Sweden | - Sweden Modem configuration |
| Switzerland | - Switzerland Modem configuration |
| Syria | - Syria Modem configuration |
| Taiwan | - Taiwan Modem configuration |
| Thailand | - Thailand Modem configuration |
| Trinidad | - Trinidad Modem configuration |
| Tunisia | - Tunisia Modem configuration |
| Turkey | - Turkey Modem configuration |
| UAE | - UAE Modem configuration |
| UK | - UK Modem configuration |
| USA/Canada | - USA/Canada Modem configuration |
| Uruguay | - Uruguay Modem configuration |
| Venezuela | - Venezuela Modem configuration |
| Yemen | - Yemen Modem configuration |

## Usage Examples

The following example specifies to use the **USA/Cananda** modem configuration.

(config)**#modem countrycode USA/Canada**

# monitor session <*number*>

Use the **monitor session** command to configure a port mirroring session. Use the **no** form of this command to remove a port mirroring session or to remove a source or destination interface. Variations of this command include:

**monitor session** <*number*> **destination interface** <*interface*> **no-isolate**
**monitor session** <*number*> **destination interface** <*interface*> **no-tag**
**monitor session** <*number*> **destination interface** <*interface*> **no-isolate no-tag**
**monitor session** <*number*> **destination interface** <*interface*> **no-tag no-isolate**
**monitor session** <*number*> **source interface** <*interface*>
**monitor session** <*number*> **source interface** <*interface*> **both**
**monitor session** <*number*> **source interface** <*interface*> **rx**
**monitor session** <*number*> **source interface** <*interface*> **tx**

## Syntax Description

| | |
|---|---|
| <*number*> | Selects the monitor session number (only one is allowed). |
| **destination** | Selects the destination interface. |
| **source** | Selects the source interface(s). A range of interfaces is allowed. |
| **interface** <*interface*> | Specifies an interface in the format <*interface type [slot/port | slot/port.sub-interface id | interface id | interface id.sub-interface id]*>. For example, for an Ethernet interface, use **eth 0/1**. Type **monitor session** <*number*> **[destination | source] interface ?** for a complete list of valid interfaces. |
| **both** | Optional. Monitors both transmitted and received traffic. |
| **rx** | Optional. Monitors received traffic only. |
| **tx** | Optional. Monitors transmitted traffic only. |
| **no-tag** | Removes the VLAN tag that is normally appended to mirrored traffic. |
| **no-isolate** | Allows native traffic to continue to pass on the port set as the mirroring session destination. |

## Default Values

By default, traffic is monitored in both directions. Also by default, the destination port is isolated from passing native traffic.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |
| Release 13.1 | Command was expanded to include no-isolate option. |

## Usage Examples

The following example sets Ethernet 0/1 as the destination interface and adds Ethernet 0/2, Ethernet 0/3, and Ethernet 0/5 as source ports:

(config)**#monitor session 1 destination interface eth 0/1**
(config)**#monitor session 1 source interface eth 0/2-3, eth 0/5**

The following example sets Gigabit-Switchport 0/1 as the destination interface and removes the VLAN tag:

(config)**#monitor session 1 destination interface gigabit-switchport 0/1 no-tag**

The following example sets Switchport 0/1 as the source interface and monitors both transmitted and received traffic:

(config)**#monitor session 1 source interface switchport 0/1 both**

The following example sets Gigabit-Switchport 0/1, and Switchport 0/2 through Switchport 0/12 as source interfaces and monitors only received traffic:

(config)**#monitor session 1 source interface gigabit 0/1, eth 0/2-12 rx**

# port-auth default

Use the **port-auth default** command to set all global port-authentication settings to their default states.

## Syntax Description

No subcommands.

## Default Values

No default value is necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example sets all global port-authentication settings to their default states:

(config)**#port-auth default**

# port-auth max-req <*number*>

Use the **port-auth max-req** command to specify the maximum number of identity requests the authenticator will transmit before restarting the authentication process. Use the **no** form of this command to return to the default setting.

## Syntax Description

| | |
|---|---|
| <*number*> | Specifies the maximum number of authentication requests. |

## Default Values

By default, the maximum number of authentication requests is set at 2.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example sets the maximum number of authentication requests at **4**:

(config)**#port-auth max-req 4**

# port-auth re-authentication

Use the **port-auth re-authentication** command to enable re-authentication. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands necessary.

## Default Values

By default, re-authentication is disabled.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example enables re-authentication:

(config)**#port-auth reauthentication**

# port-auth timeout

Use the **port-auth timeout** command to configure various port authentication timers. Use the **no** form of this command to return to the default settings. Variations of this command include:

**port-auth timeout quiet-period** <*value*>
**port-auth timeout re-authperiod** <*value*>
**port-auth timeout tx-period** <*value*>

## Syntax Description

| | |
|---|---|
| **quiet-period** <*value*> | Specifies the amount of time the system will wait before attempting another authentication once a failure has occurred. Range is 1 to 65,535 seconds. |
| **re-authperiod** <*value*> | Specifies the amount of time between scheduled re-authentication attempts. Range is 1 to 4,294,967,295 seconds. |
| **tx-period** <*value*> | Specifies the amount of time the authenticator will wait between identity requests. Range is 1 to 65,535 seconds. |

## Default Values

By default, **quiet-period** is set to 60 seconds, **re-authperiod** is set to 3600 seconds (1 hour), and **tx-period** is set to 30 seconds.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example sets the quiet-period to **10** seconds:

(config)**#port-auth timeout quiet-period 10**

# port-channel load-balance

Use the **port-channel load-balance** command to configure port aggregation load distribution. Use the **no** form of this command to reset distribution to its default. Variations of this command include:

**port-channel load-balance dst-mac**
**port-channel load-balance src-mac**

## Syntax Description

| | |
|---|---|
| **dst-mac** | Specifies the destination MAC address. |
| **src-mac** | Specifies the source MAC address. |

## Default Values

By default, load balance is set to **src-mac**.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

During port aggregation, the port channel interface must determine on which physical port to transmit packets. With the source-address configuration, the source MAC address of the received packets is used to determine this allocation. Packets coming from a specific host always use the same physical port. Likewise, when the destination address configuration is used, packets are forwarded based on the MAC address of the destination. Packets destined for a specific host always use the same physical port.

## Usage Examples

The following example sets the load distribution to use the destination MAC address:

(config)**#port-channel load-balance dst-mac**

# power-supply shutdown automatic

Use the **power-supply shutdown automatic** command to enable the power supplies to automatically shut down when the unit temperature exceeds the maximum operating temperature. Use the **no** form of this command to disable.

## Syntax Description

No subcommands.

## Default Values

By default, this command is enabled.

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |

## Usage Examples

The following example enables the power supplies to shut down automatically if the temperature gets too high:

(config)**#power-supply shutdown automatic**

# probe

Use the **probe** command to create a probe as part of network monitoring. This command is also used to enter into the Network Monitoring Probe command set once a probe is created. A probe can be one of three types: **http-request**, **icmp-echo**, or **tcp-connect**. Each probe type has a set of commands used for configuration. These additional commands are covered in *Network Monitor Probe Command Set* on page 1682. Use the **no** form of this command to delete the probe. Variations of this command include:

**probe** <*name*> **http-request**
**probe** <*name*> **icmp-echo**
**probe** <*name*> **tcp-connect**

*Issue the **no shutdown** command to activate the probe once it is configured.*

*The probe is not operational until tolerance is defined. Refer to Network Monitor Probe Command Set on page 1682* for more information.

## Syntax Description

| | |
|---|---|
| <*name*> | Specifies the name of the probe being created or indicates the probe affected by the commands that follow. |
| **http-request** | Specifies the probe type being created as an HTTP request. |
| **icmp-echo** | Specifies the probe type being created as an ICMP echo. |
| **tcp-connect** | Specifies the probe type being created as a TCP connect. |

## Default Values

By default, there are no probes configured.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example creates an ICMP echo probe called **probe1**:

```
>enable
#configure terminal
(config)#interface probe probe1 icmp-echo
(config-probe-probe1)#
```

## Technology Review

Probes are stand alone objects which help determine the status of a route based on the success or failure of probe traffic across the path. The probes can be configured to trigger at particular intervals. There are three types of probes supported by AOS: icmp-echo, tcp-connect, and http-request. Commands common to all the probe types are identified in the following section as well as isolated commands that only apply to the specific probe types.

Additional configuration commands are available for associating tracks with each probe. These are explained in the *Network Monitor Track Configuration Command Set* on page 1698.

# qos cos-map *<cos queue id> <cos value>*

Use the **qos cos-map** command to associate class of service (CoS) values with each queue. Use the **no** form of this command to return to the default settings.

## Syntax Description

| | |
|---|---|
| *<cos queue id>* | Specifies the queue number to which you are assigning CoS value(s). |
| *<cos value>* | Associates listed CoS values with a particular priority queue. Multiple Cos values can be applied to a specified queue. Valid range is 0 to 7. |

## Default Values

By default, CoS 0 and 1 are mapped to queue 1; CoS 2 and 3 are mapped to queue 2; CoS 4 and 5 are mapped to queue 3; CoS 6 and 7 are mapped to queue 4.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following example maps CoS values 4 and 5 to queue 1:

(config)**#qos cos-map 1 4 5**

# qos dscp-cos

Use the **qos dscp-cos** command to set the Differentiated Services Codepoint (DSCP) to class of service (CoS) map and enable the mapping process. Use the **no** form of this command to disable mapping. Variations of this command include:

**qos dscp-cos** *<dscp value>* **to** *<cos value>*
**qos dscp-cos default**

## Syntax Description

| | |
|---|---|
| *<dscp value>* | Specifies DSCP values (separating multiple values with a space). Valid range is 0 to 63. |
| *<cos value>* | Specifies CoS values (separating multiple values with a space). Valid range is 0 to 7. |
| **default** | Sets the map to the following default values: |

| DSCP | 0 | 8 | 16 | 24 | 32 | 40 | 48 | 56 |
|---|---|---|---|---|---|---|---|---|
| CoS | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |

## Functional Notes

When one of the specified DSCP values is detected in an incoming packet, the CoS priority is altered based on the corresponding map value. By configuring the list, the mapping functionality is enabled.

## Usage Examples

The following example enables the mapping of DSCP values 24 and 48 to CoS values 1 and 2:

(config)**#qos dscp-cos 24 48 to 1 2**

The following example disables DSCP-to-CoS mapping:

(config)**#no qos dscp-cos**

# qos map <*name*> <*number*>

Use the **qos map** command to activate the QoS Map Command Set (which allows you to create and/or edit a QoS map). For details on specific commands, refer to the section *Quality of Service (QoS) Map Command Set* on page 1984. Use the **no** form of this command to delete a map entry.

## Syntax Description

| | |
|---|---|
| <*name*> | Specifies the QoS map name. |
| <*number*> | Specifies a number to differentiate this QoS map and to assign match order. Valid range is 0 to 65,535. |

## Default Values

No default value is necessary for this command.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Functional Notes

A QoS policy is defined using a QoS map. The QoS map is a named list with sequenced entries. An entry contains a single match reference and one or more actions (**priority**, **set**, or **both**). Multiple map entries for the same QoS map are differentiated by a sequence number. The sequence number is used to assign match order.

Once created, a QoS map must be applied to an interface (using the **qos-policy out** <*map-name*> command) in order to actively process traffic. Any traffic for the interface that is not sent to the priority queue is sent using the default queuing method for the interface (such as weighted fair queuing).

## Usage Examples

The following example demonstrates basic settings for a QoS map and assigns a map to the Frame Relay interface:

```
>enable
#config terminal
(config)#qos map VOICEMAP 10
(config-qos-map)#match precedence 5
(config-qos-map)#priority 512
(config-qos-map)#exit
(config)#interface fr 1
(config-fr 1)#qos-policy out VOICEMAP
```

# qos queue-type strict-priority

Use the **qos queue-type strict-priority** command to enable queuing based strictly on the priority of each queue. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, the queue type is weighted round robin (WRR).

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following example enables strict-priority queuing:

(config)**#qos queue-type strict-priority**

# qos queue-type wrr

Use the **qos queue-type wrr** command to set weights for up to four queues. Use the **no** form of this command to set all queues to be weighted round robin (WRR). Variations of this command include:

**qos queue-type wrr** <*weight1*> <*weight2*> <*weight3*> **expedite**
**qos queue-type wrr** <*weight1*> <*weight2*> <*weight3*> <*weight4*>

## Syntax Description

| | |
|---|---|
| <*weight1-4*> | Sets the weight of each queue (up to four). All queue weights must be greater than zero except for the weight for the last queue (queue 4). The range for queues 1 to 3 is 1 to 255. The range for queue 4 is 0 to 255. |
| **expedite** | The queue 4 entry can be replaced by the **expedite** command. If set to **expedite**, then it becomes a high-priority queue. All outbound traffic is transmitted on an expedite queue prior to any other traffic in other queues. |

## Default Values

By default, all four weights are set to 25.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

The actual weight is a calculated value based on the sum of all entered weights. It is the ratio of the individual weight over the sum of all weights.

For example:

If the user enters 10, 20, 30, and 40 as the weight values, the first queue will have a ratio of 1/10. This is derived from the formula 10/(10+20+30+40). Therefore, this queue will transmit 1 packet out of every 10 opportunities.

## Usage Examples

The following example configures weights for all four queues:

(config)#**qos queue-type wrr 10 20 30 40**

# radius-server

Use the **radius-server** command to configure several global RADIUS parameters. Most of these global defaults can be overridden on a per-server basis. Use the **no** form of this command to return to the default settings. Variations of this command include the following:

**radius-server challenge-noecho**
**radius-server deadtime** <*value*>
**radius-server enable-username** <*name*>
**radius-server key** <*key*>
**radius-server retry** <*number*>
**radius-server timeout** <*value*>

## Syntax Description

| | |
|---|---|
| **challenge-noecho** | Turns off echoing of user challenge-entry. When echo is turned on, users see the text of the challenge as they type responses. Enabling this option hides the text as it is being entered. |
| **deadtime** <*value*> | Specifies how long (in minutes) a RADIUS server is considered dead once a timeout occurs. The server will not be tried again until after the deadtime expires. Valid range is 0 to 1440 minutes. |
| **enable-username** <*name*> | Specifies a user name to be used for Enable mode authentication. |
| **key** <*key*> | Specifies the shared key to use with a RADIUS server. |
| **retry** <*number*> | Specifies the number of attempts to make on a RADIUS server before marking it dead. |
| **timeout** <*value*> | Specifies how long (in seconds) to wait for a RADIUS server to respond to a request. Valid range is 1 to 1000 seconds. |

## Default Values

| | |
|---|---|
| **challenge-noecho** | Echo is turned on |
| **deadtime** | 1 minute |
| **key** | No default |
| **retry** | 3 attempts |
| **timeout** | 5 seconds |
| **enable-username** | $enab15$ |

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |
| Release 7.1 | Added **enable-username** selection. |

## Functional Notes

RADIUS servers (as defined with the **radius-server** command) may have many optional parameters. However, they are uniquely identified by their addresses and ports. Port values default to 1812 and 1813 for authorization and accounting, respectively. If a server is added to a named group but is not defined by a **radius-server** command, the server is simply ignored when accessed. Empty server lists are not allowed. When the last server is removed from a list, the list is automatically deleted.

## Usage Examples

The following example shows a typical configuration of these parameters:

(config)**#radius-server challenge-noecho**
(config)**#radius-server deadtime 10**
(config)**#radius-server timeout 2**
(config)**#radius-server retry 4**
(config)**#radius-server key my secret key**

# radius-server host

Use the **radius-server host** to specify the parameters for a remote RADIUS server. At a minimum, the address (IP or DNS name) of the server must be given. The other parameters are also allowed and (if not specified) will take default values or fall back on the global RADIUS server's default settings. Use the **no** form of this command to return to the default settings. Variations of this command include:

**radius-server host** <*ip address*>
**radius-server host** <*ip address*> **acct-port** <*port*>
**radius-server host** <*ip address*> **acct-port** <*port*> **acct-port**
**radius-server host** <*ip address*> **auth-port** <*port*>
**radius-server host** <*ip address*> **auth-port** <*port*> **acct-port**
**radius-server host** <*ip address*> **key encrypted** <*key*>
**radius-server host** <*ip address*> **key encrypted** <*key*> **acct-port**
**radius-server host** <*ip address*> **key** <*key*>
**radius-server host** <*ip address*> **key** <*key*> **acct-port**
**radius-server host** <*ip address*> **retransmit** <*number*>
**radius-server host** <*ip address*> **retransmit** <*number*> **acct-port**
**radius-server host** <*ip address*> **timeout** <*value*>
**radius-server host** <*ip address*> **timeout** <*value*> **acct-port**

## Syntax Description

| | |
|---|---|
| <*ip address*> | Specifies a valid IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| **acct-port** <*port*> | Sends accounting requests to this remote port. Choose a port value between 0 and 65,535. |
| **auth-port** <*port*> | Sends authentication requests to this remote port. Choose a port value between 0 and 65,535. |
| **key** <*key*> | Defines the shared key with the RADIUS server (uses RADIUS global setting if not given). Note that the key must appear last on the input line since it reads the rest of the line beyond the **key** keyword. |
| **key encrypted** <*key*> | Defines an encrypted shared key with the RADIUS server (uses RADIUS global setting if not given). Note that the key must appear last on the input line since it reads the rest of the line beyond the **key** keyword. |
| **retransmit** <*number*> | Retries server after timeout this number of times (uses RADIUS global setting if not given). Range is 1 to 100. |
| **timeout** <*value*> | Waits for a response this number of seconds (uses RADIUS global setting if not given). Range is 1 to 1000 seconds. |

## Default Values

By default, **acct-port** is set to 1813 and **auth-port** is set to 1812.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |
| Release 11.1 | Command was expanded to include the **key encrypted** command. |

## Usage Examples

The following example shows a typical configuration of these parameters:

(config)**#radius-server host 1.2.3.4**
(config)**#radius-server host 3.3.1.2 acct-port 1646 key my key**

# route-map

Use the **route-map** command to create a route map and enter the Route Map Configuration command set. A route map is a type of filter that matches various attributes and then performs actions on the way the route is redistributed. Use the **no** form of this command to delete a route map. Variations of this command include:

**route-map** *<name>* *<number>*
**route-map** *<name>* **deny** *<number>*
**route-map** *<name>* **permit** *<number>*

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies a name for the route map. |
| **deny** | Specifies not to redistribute routes matching the route map attributes. |
| **permit** | Redistributes routes matching the route map attributes. |
| *<number>* | Specifies a sequence number of this route entry. Range is 1 to 4,294,967,295. |

## Default Values

By default, no route maps are defined.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Functional Notes

Route maps can be assigned to a neighbor using the **route-map** command in the BGP Neighbor command set. Refer to *route-map <name>* on page 1675 for more information.

## Usage Examples

The following example creates the route map, specifies that routes matching its criteria will be denied, and assigns a sequence number of 100:

(config)**#route-map MyMap deny 100**
(config-route-map)#

You can then define the attributes of the route map from the Route Map Configuration Command set. Enter a **?** at the **(config-route-map)#** prompt to explore the available options.

# router bgp

Use the **router bgp** command to enter the BGP Configuration mode. Refer to the *BGP Configuration Command Set* on page 1649 for more information. Use the **no** form of this command to disable BGP routing.

## Syntax Description

No subcommands.

## Default Values

No default values necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example uses the **router bgp** command to enter the BGP Configuration mode:

```
(config)#router bgp
(config-bgp)#
```

## Technology Review

The following AOS BGP-related guidelines may help guide decisions made during basic BGP implementation.

Ignore route if next hop is unreachable.

Prefer route with largest weight (only used in the local router, set by applying route maps to set this value on desired inbound updates).

Prefer route with largest local preference.

Prefer route injected by this router via network command.

Prefer route with shortest AS_PATH.

Prefer route with lowest origin type. Routes originally injected by the network command or aggregation (IGP) have a lower origin type than those originally injected by redistribution into BGP.

Prefer routes with lowest MED value.

Before the route is installed into the route table (forwarding table), a check is made of other sources that may have information about the same subnet (static routes, IGP, etc.) The route with the lowest administrative distance is installed.

# router ospf

Use the **router ospf** command to activate OSPF in the router and to enter the OSPF Configuration mode. Refer to the section *Router (OSPF) Configuration Command Set* on page 1703 for more information. Use the **no** form of this command to disable OSPF routing.

## Syntax Description

No subcommands.

## Default Values

By default, OSPF is disabled.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Functional Notes

The AOS can be configured to use OSPF with the firewall enabled (using the **ip firewall** command). To do this, configure the OSPF networks as usual, specifying which networks the system will listen for and broadcast OSPF packets to. Refer to *ip firewall* on page 514 for more information.

To apply stateful inspection to packets coming into the system, create a policy class that describes the type of action desired and then associate that policy class to the particular interface (refer to *ip policy-class <name>* on page 549). The firewall is intelligent and will only allow OSPF packets that were received on an OSPF configured interface. No modification to the policy class is required to allow OSPF packets into the system.

## Usage Examples

The following example uses the **router ospf** command to enter the OSPF Configuration mode:

(config)**#router ospf**

# router pim-sparse

Use the **router pim-sparse** command to globally enable protocol-independent multicast (PIM) on the unit and to enter the PIM Sparse Configuration mode. Use the **no** form of this command to disable PIM Sparse routing. Refer to the section *Router (PIM Sparse) Configuration Command Set* on page 1718 for more information on the subcommands for PIM Sparse Configuration mode.

## Syntax Description

No subcommands.

## Default Values

No default values necessary for this command.

## Functional Notes

Additional commands for PIM are found in the related interface configuration modes. Refer to the **ip pim-sparse** commands in sections such as *Ethernet Interface Configuration Command Set* on page 819, *Frame Relay Sub-Interface Config Command Set* on page 1175, *HDLC Interface Configuration Command Set* on page 1249, *Loopback Interface Configuration Command Set* on page 1315, *PPP Interface Configuration Command Set* on page 1379, *Tunnel Configuration Command Set* on page 1463, and *VLAN Interface Configuration Command Set* on page 1542 for more information.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example uses the **router pim-sparse** command to enter the PIM Sparse Configuration mode:

```
(config)#router pim-sparse
(config-pim-sparse)#
```

# router rip

Use the **router rip** command to enter the RIP Configuration mode. Use the **no** form of this command to disable RIP routing. Refer to the section *Router (RIP) Configuration Command Set* on page 1722 for more information.

## Syntax Description

No subcommands.

## Default Values

No default values necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example uses the **router rip** command to enter the RIP Configuration mode:

```
(config)#router rip
(config-rip)#
```

## Technology Review

The RIP protocol is based on the Bellham-Ford (distance-vector) algorithm. This algorithm provides that a network will converge to the correct set of shortest routes in a finite amount of time, provided that:

Gateways continuously update their estimates of routes.

Updates are not overly delayed and are made on a regular basis.

The radius of the network is not excessive.

No further topology changes take place.

RIP is described in RFC 1058 (Version 1) and updated in RFCs 1721, 1722, and 1723 for Version 2. Version 2 includes components that ease compatibility in networks operating with RIP V1.

All advertisements occur on regular intervals (every 30 seconds). Normally, a route that is not updated for 180 seconds is considered dead. If no other update occurs in the next 60 seconds for a new and better route, the route is flushed after 240 seconds. Consider a connected route (one on a local interface). If the interface fails, an update is immediately triggered for that route only (advertised with a metric of 16).

Now consider a route that was learned and does not receive an update for 180 seconds. The route is marked for deletion, and even if it was learned on an interface, a poisoned (metric equals 16) route should be sent by itself immediately and during the next two update cycles with the remaining normal split horizon update routes. Following actual deletion, the poison reverse update ceases. If an update for a learned route is not received for 180 seconds, the route is marked for deletion. At that point, a 120-second garbage collection (GC) timer is started. During the GC timer period, expiration updates are sent with the metric for the timed-out route set to 16.

If an attached interface goes down, the associated route is immediately (within the same random five-second interval) triggered. The next regular update excludes the failed interface. This is the so-called first hand knowledge rule. If a gateway has first hand knowledge of a route failure (connected interfaces) or reestablishment, the same action is taken. A triggered update occurs, advertising the route as failed (metric equals 16) or up (normal metric) followed by the normal scheduled update.

The assumption here is that if a gateway missed the triggered update, it will eventually learn from another gateway in the standard convergence process. This conserves bandwidth.

RIP-Related Definitions:

| | |
|---|---|
| Route | A description of the path and its cost to a network. |
| Gateway | A device that implements all or part of RIP (a router). |
| Hop | A metric that provides the integer distance (number of intervening gateways) to a destination network gateway. |
| Advertisement | A broadcast or multicast packet to port 520 that indicates the route for a given destination network. |
| Update | An advertisement sent on a regular 30-second interval including all routes exclusive of those learned on an interface. |

# service password-encryption

Use the **service password-encryption** command to turn on global password protection. Use the **no** form of this command to return to default settings.


*If you need to go back to a previous revision of the code (e.g., AOS Revision 10), this command must be disabled first. Once the service is disabled, all necessary passwords must be re-entered so that they are in the clear text form. If this is not done properly, you will not be able to log back in to the unit after you revert to a previous revision that does not support password encryption.*

## Syntax Description

No subcommands.

## Default Values

By default, global password protection is disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Functional Notes

When enabled, all currently configured passwords are encrypted. Also, any new passwords are encrypted after they are entered. Password encryption is applied to all passwords, including passwords for user name, Enable mode, Telnet/console, PPP, BGP, and authentication keys. When passwords are encrypted, unauthorized persons cannot view them in configuration files since the encrypted form of the password is displayed in the running-config. While this provides some level of security, the encryption method used with password encryption is not a strong form of encryption so you should take additional network security measures.


*You cannot recover a lost encrypted password. You must erase the startup-config and set a new password.*

## Usage Examples

The following example enables password encryption for all passwords on the unit:

(config)**#service password-encryption**

# snmp-server chassis-id <*string*>

Use the **snmp-server chassis-id** command to specify an identifier for the Simple Network Management Protocol (SNMP) server. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*string*> | Identifies the product using an alphanumeric string (up to 32 characters in length). |

## Default Values

By default, the **snmp-server chassis-id** is set to **Chassis ID**.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Usage Examples

The following example configures a chassis ID of **A432692**:

(config)**#snmp-server chassis-id A432692**

# snmp-server community

Use the **snmp-server community** command to specify a community string to control access to the Simple Network Management Protocol (SNMP) information. Use the **no** form of this command to remove a specified community. Variations of this command include:

**snmp-server community** <*community*>
**snmp-server community** <*community*> <*listname*>
**snmp-server community** <*community*> **ro**
**snmp-server community** <*community*> **ro** <*listname*>
**snmp-server community** <*community*> **rw**
**snmp-server community** <*community*> **rw** <*listname*>
**snmp-server community** <*community*> **view** <*viewname*>
**snmp-server community** <*community*> **view** <*viewname*> <*listname*>
**snmp-server community** <*community*> **view** <*viewname*> **ro**
**snmp-server community** <*community*> **view** <*viewname*> **ro** <*listname*>
**snmp-server community** <*community*> **view** <*viewname*> **rw**
**snmp-server community** <*community*> **view** <*viewname*> **rw** <*listname*>

## Syntax Description

| | |
|---|---|
| <*community*> | Specifies the community string (a password to grant SNMP access). |
| **view** <*viewname*> | Optional. Specifies a previously defined view. Views define objects available to the community. For information on creating a new view, refer to *snmp-server view <name> <value>* on page 672. |
| **ro** | Optional. Grants read-only access, allowing retrieval of MIB objects. |
| **rw** | Optional. Grants read-write access, allowing retrieval and modification of MIB objects. |
| <*listname*> | Optional. Specifies an access-control list name used to limit access. Refer to *ip access-list extended <name>* on page 492 and *ip access-list standard <name>* on page 496 for more information on creating access-control lists. |

## Default Values

By default, there are no configured SNMP communities.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 9.1 | Command was expanded to include the **view** option. |

## Usage Examples

The following example specifies a community named **MyCommunity**, specifies a previously defined view named **blockinterfaces**, and assigns read-write access:

(config)**#snmp-server community MyCommunity view blockinterfaces rw**

# snmp-server contact

Use the **snmp-server contact** command to specify SNMP server contact information. Use the **no** form of this command to remove a configured contact. Variations of this command include:

**snmp-server contact email** <*address*>
**snmp-server contact pager** <*number*>
**snmp-server contact phone** <*number*>
**snmp-server contact** <*"string"*>

## Syntax Description

| | |
|---|---|
| **email** <*address*> | Specifies email address for the SNMP server contact. |
| **pager** <*number*> | Specifies pager number for the SNMP server contact. |
| **phone** <*number*> | Specifies phone number for the SNMP server contact. |
| <*"string"*> | Populates the sysContact string using an alphanumeric string enclosed in quotation marks (up to 32 characters in length). |

## Default Values

No default values necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example specifies **6536999** for the pager number:

(config)**#snmp-server contact pager 6536999**

# snmp-server enable traps

Use the **snmp-server enable traps** command to enable all Simple Network Management Protocol (SNMP) traps available on your system. Use the **no** form of this command to disable SNMP traps. Variations of this command include:

**snmp-server enable traps**
**snmp-server enable traps snmp**

## Syntax Description

| | |
|---|---|
| **snmp** | Optional. Specifies the type of notification trap to enable, which at this time only includes SNMP.<br>The following traps are supported:<br>**coldStart**<br>**warmStart**<br>**linkUp**<br>**linkDown**<br>**authenticationFailure** |

## Default Values

By default, there are no enabled traps.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

This command was written to include additional traps at a later time. Currently, snmp is the only available option. Therefore, issuing either **snmp-server enable traps** or **snmp-server enable traps snmp** will enable snmp traps.

## Usage Examples

The following example enables SNMP traps:

(config)**#snmp-server enable traps snmp**

# snmp-server group

Use the **snmp-server group** command to specify a new Simple Network Management Protocol (SNMP) group to control access to SNMP information. Use the **no** form of this command to remove a specified group. Variations of this command include:

**snmp-server group** <*groupname*> **v1 read** <*name*> **write** <*name*> **notify** <*name*> **access** <*listname*>
**snmp-server group** <*groupname*> **v2c read** <*name*> **write** <*name*> **notify** <*name*> **access** <*listname*>
**snmp-server group** <*groupname*> **v3 auth read** <*name*> **write** <*name*> **notify** <*name*>
**access** <*listname*>
**snmp-server group** <*groupname*> **v3 noauth read** <*name*> **write** <*name*> **notify** <*name*>
**access** <*listname*>
**snmp-server group** <*groupname*> **v3 priv read** <*name*> **write** <*name*> **notify** <*name*>
**access** <*listname*>

## Syntax Description

| | |
|---|---|
| **access** <*listname*> | Specifies an access control list entry. |
| <*groupname*> | Specifies the name of the SNMP group (32 characters maximum). |
| **notify** <*name*> | Specifies a notify-view entry (32 characters maximum). |
| **read** <*name*> | Specifies a read-view entry (32 characters maximum). |
| **write** <*name*> | Specifies a write-view entry (32 characters maximum). |
| **v1** | Uses SNMP version 1 security model. |
| **v2c** | Uses SNMP version 2c security model. |
| **v3** | Uses SNMP version 3 user-based security model (USM) |
| | Options if v3 is used: |
| | **auth** Indicates that authentication is used. |
| | **noauth** Indicates that no authentication is used. |
| | **priv** Indicates that privacy authentication is used. |

## Default Values

If no views are specified, the system automatically assigns default read- and notify-views that have no restrictions.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example defines a group called **securityV3auth** using version 3 security model, authentication, and no access control list to verify:

(config)**#snmp-server group securityV3auth v3 auth**

# snmp-server host

Use the **snmp-server host** command to configure the host to receive Simple Network Management Protocol (SNMP) notifications. Use the **no** form of this command to remove a specified host. Variations of this command include the following:

**snmp-server host** <*ip address*> **informs version 1** <*community*>
**snmp-server host** <*ip address*> **informs version 1** <*community*> **snmp**
**snmp-server host** <*ip address*> **informs version 2c** <*community*>
**snmp-server host** <*ip address*> **informs version 2c** <*community*> **snmp**
**snmp-server host** <*ip address*> **informs version 3 auth** <*community*>
**snmp-server host** <*ip address*> **informs version 3 auth** <*community*> **snmp**
**snmp-server host** <*ip address*> **informs version 3 noauth** <*community*>
**snmp-server host** <*ip address*> **informs version 3 noauth** <*community*> **snmp**
**snmp-server host** <*ip address*> **informs version 3 priv** <*community*>
**snmp-server host** <*ip address*> **informs version 3 priv** <*community*> **snmp**
**snmp-server host** <*ip address*> **traps** <*community*>
**snmp-server host** <*ip address*> **traps version 1** <*community*>
**snmp-server host** <*ip address*> **traps version 1** <*community*> **snmp**
**snmp-server host** <*ip address*> **traps version 2c** <*community*>
**snmp-server host** <*ip address*> **traps version 2c** <*community*> **snmp**
**snmp-server host** <*ip address*> **traps version 3 auth** <*community*>
**snmp-server host** <*ip address*> **traps version 3 auth** <*community*> **snmp**
**snmp-server host** <*ip address*> **traps version 3 noauth** <*community*>
**snmp-server host** <*ip address*> **traps version 3 noauth** <*community*> **snmp**
**snmp-server host** <*ip address*> **traps version 3 priv** <*community*>
**snmp-server host** <*ip address*> **traps version 3 priv** <*community*> **snmp**

## Syntax Description

| | |
|---|---|
| <*ip address*> | Specifies the IP address of the SNMP host that receives the SNMP information. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| **informs** | Enables informs to this host.<br>The following traps are supported:<br>**coldStart**<br>**warmStart**<br>**linkUp**<br>**linkDown**<br>**authenticationFailure** |
| <*community*> | Specifies the community string (used as a password) (16 characters maximum) for authorized agents to obtain access to SNMP information. |
| **traps** | Enables traps to this host. If the version is not specified, version 1 is used. |

| | |
|---|---|
| **version 1** | Uses SNMP version 1 security model. |
| **version 2c** | Uses SNMP version 2c security model. |
| **version 3** | Uses SNMP version 3 user-based security model (USM) |
| **snmp** | Optional. Enables a subset of traps specified in RFC1157. |

## Default Values

No default is necessary with this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 13.1 | Command was expanded to include informs options. |

## Usage Examples

The following example sends all SNMP traps to the host at address **190.3.44.69** and community string **MyCommunity** using SNMP version 2c:

(config)**#snmp-server host 190.3.44.69 traps version 2c MyCommunity snmp**

# snmp-server inform

Use the **snmp-server inform** retries command to set the number of retry attempts for a response and set the amount of time to wait for a response before allowing a new request. Use the **no** form of this command to return to the default setting. Variations of this command include the following:

**snmp-server inform retries** <*number*>
**snmp-server inform timeout** <*value*>

## Syntax Description

| | |
|---|---|
| **retries** <*number*> | Specifies number of retries for a response. The range is from 1 to 100. |
| **timeout** <*value*> | Specifies time (in seconds) to wait for a response. The range is from 1 to 1000 seconds. |

## Default Values

By default, the retry count is set to 3 and the timeout is set to 5 seconds.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example sets the retry count to 10:

(config)**#snmp-server inform retries 10**

# snmp-server location <*"string"*>

Use the **snmp-server location** command to specify the Simple Network Management Protocol (SNMP) system location string. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*"string"*> | Populates the system location string using an alphanumeric string enclosed in quotation marks (up to 32 characters in length). |

## Default Values

By default, the **snmp-server location** is set to **ADTRAN**.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example specifies a location of **5th Floor Network Room**:

(config)**#snmp-server location "5th Floor Network Room"**

# snmp-server management-url *<url>*

Use the **snmp-server management-url** command to specify the URL for the device's management software. Use the **no** form of this command to remove the management URL.

## Syntax Description

| | |
|---|---|
| *<url>* | Specifies the URL for the management software. |

## Default Values

No default is necessary for this command.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Usage Examples

The following example specifies the URL http://www.mywatch.com as the device's management software:

(config)**#snmp-server management-url http://www.mywatch.com**

# snmp-server management-url-label *<label>*

Use the **snmp-server management-url-label** command to specify a label for the URL of the device's management software. Use the **no** form of this command to remove the label.

## Syntax Description

| | |
|---|---|
| *<label>* | Specifies a label for the URL of the management software (maximum length 255 characters). |

## Default Values

No default is necessary for this command.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Usage Examples

The following example specifies the label **watch** for the management software:

(config)**#snmp-server management-url-label watch**

# snmp-server source-interface *<interface>*

Use the **snmp-server source-interface** command to specify a source interface for SNMP traffic (including traps and get/set requests) originated by the unit. The IP address of the specified interface will be used to source all SNMP traffic. Use the **no** form of this command to remove the specified interface.

## Syntax Description

| | |
|---|---|
| *<interface>* | Specifies the interface that should originate SNMP traffic. Specify an interface in the format *<interface type [slot/port \| slot/port.sub-interface id \| interface id \| interface id.sub-interface id]>*. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **snmp-server source-interface ?** for a complete list of valid interfaces. |

## Default Values

By default, there is no source-interface defined.

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |

## Usage Examples

The following example specifies the **ethernet 0/1** should be the source for all SNMP traps and get/set requests:

(config)**#snmp-server source-interface ethernet 0/1**

## snmp-server user

Use the **snmp-server user** command to configure Simple Network Management Protocol (SNMP) users to control access to SNMP information. Use the **no** form of this command to remove a specified host. Variations of this command include:

**snmp-server user** <*username*> <*groupname*> **v1**
**snmp-server user** <*username*> <*groupname*> **v1 access** <*listname*>
**snmp-server user** <*username*> <*groupname*> **v2c**
**snmp-server user** <*username*> <*groupname*> **v2c access** <*listname*>
**snmp-server user** <*username*> <*groupname*> **v3**
**snmp-server user** <*username*> <*groupname*> **v3 access** <*listname*>
**snmp-server user** <*username*> <*groupname*> **v3 auth md5** <*password*>
**snmp-server user** <*username*> <*groupname*> **v3 auth md5** <*password*> **access** <*listname*>
**snmp-server user** <*username*> <*groupname*> **v3 auth md5** <*password*> **priv des** <*password*>
**snmp-server user** <*username*> <*groupname*> **v3 auth md5** <*password*> **priv des** <*password*> **access** <*listname*>
**snmp-server user** <*username*> <*groupname*> **v3 auth sha** <*password*>
**snmp-server user** <*username*> <*groupname*> **v3 auth sha** <*password*> **access** <*listname*>
**snmp-server user** <*username*> <*groupname*> **v3 auth sha** <*password*> **priv des** <*password*>
**snmp-server user** <*username*> <*groupname*> **v3 auth sha** <*password*> **priv des** <*password*> **access** <*listname*>


*If password encryption is used, the running-config changes to include the key-word* ***encrypted*** *before each password entry. See the* ***service password-encryption*** *command.*

### Syntax Description

| | | |
|---|---|---|
| **access** <*listname*> | Specifies an access control list entry. | |
| <*groupname*> | Specifies the name of the group the user belongs to. | |
| <*username*> | Specifies the name of the user. | |
| **v1** | Uses SNMP version 1 security model. | |
| **v2c** | Uses SNMP version 2c security model. | |
| **v3** | Uses SNMP version 3 (user-based security model). | |
| | Options if v3 is used: | |
| | **auth** | Indicates that authentication is used. |
| | **des56** | Use the CBC-DES privacy authentication algorithm. |
| | **md5** | Uses the HMAC-MD5-96 authentication level. |
| | **priv** | Indicates that privacy authentication is used. |
| | **sha** | Uses the HMAC-SHA-96 authentication level. |

| | | |
|---|---|---|
| | **auth-password** | A password string used to build the key for the authentication level. |
| | **priv-password** | A password string used for data encryption between the host and agent. |
| *<password>* | Indicates a password entry. | |

## Default Values

No default is necessary with this command.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example enters a new user named **BobbyW** and assigns the user to a group called **securityV3auth** using version 3 security model with authentication method md5 with a password of **passWORD6243** and no access control list to verify:

(config)**#snmp-server user BobbyW securityV3auth v3 auth md5 passWORD6243**

# snmp-server view *<name> <value>*

Use the **snmp-server view** command to create or modify a Simple Network Management Protocol (SNMP) view entry. Use the **no** form of this command to remove an entry. Variations of this command include:

**snmp-server view** *<name> <value>* **excluded**
**snmp-server view** *<name> <value>* **included**

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies a label for the view record being created. The name is a record reference. |
| *<value>* | Specifies the object identifier (OID) to include or exclude from the view. To identify the subtree, specify a string using numbers, such as 1.4.2.6.8. Replace a single subidentifier with the asterisk (*) to specify a subtree family. |
| **excluded** | Specifies a view to be excluded. |
| **included** | Specifies a view to be included. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Usage Examples

The **snmp-server view** command can include or exclude a group of OIDs. The following example shows how to create a view (named **blockInterfaces**) to exclude the OID subtree family 1.3.3.1.2.1.2:

(config)**#snmp-server view blockInterfaces 1.3.6.1.2.1.2.* excluded**

The following example shows how to create a view (named **block**) to include a specific OID:

(config)**#snmp-server view block 1.3.6.1.2.1.2. included**

# sntp retry-timeout <*value*>

Use the **sntp retry-timeout** command to set the amount of time to wait for a response before allowing a new request. Use the **no** form of this command to return to the default setting.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies time (in seconds) to wait for a response before retrying. The range is from 3 to 4,294,967,294 seconds. |

## Default Values

By default, the retry timeout is set to 5 seconds.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Usage Examples

The following example sets the SNTP retry timeout to 10 seconds:

(config)**#sntp retry-time 10**

# sntp server

Use the **sntp server** command to set the host name of the SNTP server as well as the version of SNTP to use. The Simple Network Time Protocol (SNTP) is an abbreviated version of the Network Time Protocol (NTP). SNTP is used to set the time of the AOS product over a network. The SNTP server usually serves the time to many devices within a network. Use the **no** form of this command to return to the default setting. Variations of this command include:

**sntp server version** <*hostname*>
**sntp server version** <*ip address*>
**sntp server version** <*number*>

## Syntax Description

| | |
|---|---|
| <*hostname*> | Specifies the host name of the SNTP server. |
| <*ip address*> | Specifies the IP address of the SNTP server. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| **version** <*number*> | Optional. Specifies which NTP version is used. Valid range is 1 to 3. |

## Default Values

By default, NTP version is set to 1.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Usage Examples

The following example sets the SNTP server to **time.nist.gov** using SNTP version 1 (the default version):

(config)**#sntp server time.nist.gov**

The following example sets the SNTP server as **time.nist.gov**. All requests for time use version 2 of the SNTP:

(config)**#sntp server time.nist.gov version 2**

# sntp wait-time <*value*>

Use the **sntp wait-time** command to set the time between updates from the time server. Use the **no** form of this command to return to the default setting.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies time (in seconds) between updates. Range is 10 to 4,294,967,294 seconds. |

## Default Values

By default, the wait time is set to 86,400 seconds (1 day).

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Usage Examples

The following example sets the SNTP wait time to two days:

(config)**#sntp wait-time 172800**

# spanning-tree edgeport bpdufilter default

Use the **spanning-tree edgeport bpdufilter default** command to enable the BPDU filter on all ports by default. Use the **no** form of this command to disable the setting.

## Syntax Description

No subcommands.

## Default Values

By default, **spanning-tree edgeport bpdufilter default** is disabled.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

The BPDU filter blocks any BPDUs from being transmitted and received on an interface. This can be overridden on an individual port.

## Usage Examples

The following example enables the bpdufilter on all ports by default:

(config)**#spanning-tree edgeport bpdufilter default**

To disable the BPDU filter on a specific interface, issue the appropriate commands for the given interface using the following commands as an example:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#spanning-tree bpdufilter disable**

# spanning-tree edgeport bpduguard default

Use the **spanning-tree edgeport bpduguard default** command to enable the BPDU guard on all ports by default. Use the **no** form of this command to disable the setting.

## Syntax Description

No subcommands.

## Default Values

Disabled by default.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

The bpduguard blocks any BPDUs from being received on an interface. This can be overridden on an individual port.

## Usage Examples

The following example enables the BPDU guard on all ports by default.

(config)**#spanning-tree bpduguard default**

To disable the BPDU guard on a specific interface, issue the appropriate commands for the given interface using the following commands as an example:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#spanning-tree bpduguard disable**

# spanning-tree edgeport default

Use the **spanning-tree edgeport default** command to configure all ports to be edgeports by default. Use the **no** form of this command to disable the setting.

## Syntax Description

No subcommands.

## Default Values

Disabled by default.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following example configures all interfaces running spanning tree to be edgeports by default:

(config)**#spanning-tree edgeport default**

An individual interface can be configured to not be considered an edgeport. For example:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#spanning-tree edgeport disable**

or

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#no spanning-tree edgeport**

# spanning-tree forward-time <*value*>

Use the **spanning-tree forward-time** command to specify the delay interval (in seconds) when forwarding spanning-tree packets. Use the **no** form of this command to return to the default interval.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the forwarding delay interval in seconds. Range is 4 to 30 seconds. |

## Default Values

By default, the forwarding delay is set to 15 seconds.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following example sets the forwarding time to 18 seconds:

(config)**#spanning-tree forward-time 18**

# spanning-tree hello-time <*value*>

Use the **spanning-tree hello-time** command to specify the delay interval (in seconds) between hello bridge protocol data units (BPDUs). To return to the default interval, use the **no** form of this command.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the delay interval (in seconds) between hello BPDUs. Range is 0 to 1,000,000 seconds. |

## Default Values

By default, the delay is set to 2 seconds.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following example configures a **spanning-tree hello-time** interval of 10,000 seconds:

(config)**#spanning-tree hello-time 10000**

## spanning-tree max-age *<value>*

Use the **spanning-tree max-age** command to specify the interval (in seconds) the spanning tree will wait to receive Bridge Protocol Data Units (BPDUs) from the root bridge before assuming the network has changed (thus re-evaluating the spanning-tree topology). Use the **no** form of this command to return to the default interval.

### Syntax Description

| | |
|---|---|
| *<value>* | Specifies the wait interval (in seconds) between received BPDUs (from the root bridge). Range is 6 to 40 seconds. |

### Default Values

By default, the wait interval is set at 20 seconds.

### Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

### Usage Examples

The following example configures a wait interval of 45 seconds:

(config)**#spanning-tree max-age 45**

# spanning-tree mode

Use the **spanning-tree mode** command to choose a spanning-tree mode of operation. Use the **no** form of this command to return to the default setting. Variations of this command include:

**spanning-tree mode rstp**
**spanning-tree mode stp**

## Syntax Description

| | |
|---|---|
| **rstp** | Enables rapid spanning-tree protocol. |
| **stp** | Enables spanning-tree protocol. |

## Default Values

By default, **spanning-tree mode** is set to **rstp**.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following example sets the spanning-tree mode to rapid spanning-tree protocol:

(config)**#spanning-tree mode rstp**

# spanning-tree pathcost method

Use the **spanning-tree pathcost method** command to select a short or long pathcost method used by the spanning-tree protocol. Use the **no** form of this command to return to the default setting. Variations of this command include:

**spanning-tree pathcost method long**
**spanning-tree pathcost method short**

## Syntax Description

| | |
|---|---|
| **long** | Specifies a long pathcost method. |
| **short** | Specifies a short pathcost method. |

## Default Values

By default, **spanning-tree pathcost** is set to **short**.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following example specifies that the spanning-tree protocol use a long pathcost method:

(config)**#spanning-tree pathcost method long**

# spanning-tree priority <*value*>

Use the **spanning-tree priority** command to set the priority for spanning-tree interfaces. The lower the priority value, the higher the likelihood the configured spanning-tree interface will be the root for the bridge group. To return to the default bridge priority value, use the **no** version of this command.

## Syntax Description

| | |
|---|---|
| <*value*> | Sets a priority value for the bridge interface. Configuring this value to a low number increases the interface's chance of being the root. Therefore, the maximum priority level would be 0. Range is 0 to 65,535. |

## Default Values

By default, the priority level is set to 32,768.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following example sets **spanning-tree priority** to the maximum level:

(config)**#spanning-tree priority 0**

# stack

Use the **stack** command to configure switch-stacking options. Use the **no** form of these commands to disable these features. Variations of this command include:

**stack master**
**stack master** *<vlan id>*
**stack master** *<vlan id> <ip address> <subnet mask>*
**stack member** *<mac address>*
**stack member** *<mac address> <unit id>*
**stack vlan** *<vlan id>*

## Syntax Description

| | |
|---|---|
| **master** | Specifies that the unit will be the master of the stack. |
| *<vlan id>* | Specifies the VLAN ID the stack will use for communication. |
| *<ip address* | Configures the network mask of the private IP network. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| *<subnet mask>* | Specifies the subnet mask that corresponds to a range of IP addresses (network). Subnet masks can be expressed in dotted decimal notation (for example, **255.255.255.0**) or as a prefix length (for example, **/24**). |
| **member** | Adds a switch to the stack. |
| *<mac address>* | Specifies a valid 48-bit MAC address of the unit being added. MAC addresses should be expressed in following format xx:xx:xx:xx:xx:xx (for example, **00:A0:C8:00:00:01**). |
| *<unit id>* | Specifies the unit ID of the switch being added. |
| **vlan** *<vlan id* | Specifies the VLAN ID of the stack of which you are a member. |

## Default Values

By default, stack VLAN is 2386, and the stack IP network is 169.254.0.0/24.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Usage Examples

The following example configures the unit to be the stack master and use the default stack VLAN and IP network.

(config)**#stack master 2000**

The following example configures the unit to be the stack master and use VLAN 2000 as the management VLAN and 192.168.1.0/24 as the management network.

(config)**#stack master 2000 192.168.1.0 255.255.255.0**

The following example adds the switch with the CPU MAC address 00:A0:C8:00:8C:20 to the stack; also assigns the number 2 as the new stack member's unit ID.

(config)**#stack member 00:A0:C8:00:8C:20 2**

The following example specifies that this unit is in the stack using VLAN 2000 as its management VLAN; also specifies that this unit is in stack member mode (not a stack-master).

(config)**#stack vlan 2000**

# tacacs-server

Use the **tacacs-server** command to customize settings for communication with TACACS servers. Use the **no** form of this command to return to default settings. Variations of this command include the following:

**tacacs-server host [**<*hostname*> **|** <*ip address*>**]**
**tacacs-server host [**<*hostname*> **|** <*ip address*>**] key** <*key*>
**tacacs-server host [**<*hostname*> **|** <*ip address*>**] port** <*tcp port*>
**tacacs-server host [**<*hostname*> **|** <*ip address*>**] timeout** <*value*>
**tacacs-server key** <*key*>
**tacacs-server packet maxsize** <*value*>
**tacacs-server timeout** <*value*>

## Syntax Description

| | |
|---|---|
| **host [**<*hostname*> **|** <*ip address*>**]** | Specifies the IP host by name or IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| **key** <*key*> | Sets an encryption key to be used for encrypting and decrypting the traffic between the Network Access Server (NAS) and the TACACS+ daemon. Setting a key for a particular server (using the **tacacs-server host [**<*hostname*> **|** <*ip address*>**] key** <*key*> command) supersedes keys set globally using the **tacacs-server key** <*key*> command. |
| **port** <*tcp port*> | Specifies the TCP port number to be used when connecting to the TACACS+ daemon. Range is 1 to 65,535. |
| **packet maxsize** <*value*> | Specifies a maximum packet size for this server. Range is 10,240 to 65,535. |
| **timeout** <*value*> | Specifies a timeout limit (in seconds) that the unit will wait for a response from the daemon before declaring an error. Range is 1 to 1000 seconds. Setting a timeout for a particular server (using the **tacacs-server host [**<*hostname*> **|** <*ip address*>**] timeout** <*value*> command) supersedes time limits set globally using the **tacacs-server timeout** <*value*> command. |

## Default Values

By default, the key is set to **key** and the default TCP port number is **49**.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example sets a timeout limit of **60** seconds for the specified server:

(config)**#tacacs-server host 10.5.6.7 timeout 60**

## thresholds

Use the **thresholds** command to specify DS1 performance counter thresholds. Use the **no** form of this command to return to default settings. Variations of this command include:

**thresholds BES [15Min | 24Hr]** <*number*>
**thresholds CSS [15Min | 24Hr]** <*number*>
**thresholds DM [15Min | 24Hr]** <*number*>
**thresholds ES [15Min | 24Hr]** <*number*>
**thresholds LCV [15Min | 24Hr]** <*number*>
**thresholds LES [15Min | 24Hr]** <*number*>>
**thresholds PCV [15Min | 24Hr]** <*number*>
**thresholds SEFS [15Min | 24Hr]** <*number*>
**thresholds SES [15Min | 24Hr]** <*number*>
**thresholds UAS [15Min | 24Hr]** <*number*>


*Threshold settings are applied to ALL DS1s.*

### Syntax Description

| | |
|---|---|
| **BES** | Specifies the bursty errored seconds threshold. |
| **CSS** | Specifies the controlled slip seconds threshold. |
| **DM** | Specifies the degraded minutes threshold. |
| **ES** | Specifies the errored seconds threshold. |
| **LCV** | Specifies the line code violations threshold. |
| **LES** | Specifies the line errored seconds threshold. |
| **PCV** | Specifies the path coding violations threshold. |
| **SEFS** | Specifies the severely errored framing seconds threshold. |
| **SES** | Specifies the severely errored seconds threshold. |
| **UAS** | Specifies the unavailable seconds threshold. |
| **15Min** | Specifies that the threshold you are setting is for the counter's 15 minute statistics. |
| **24Hr** | Specifies that the threshold you are setting is for the counter's 24 hour statistics. |
| <*number*> | Specifies the maximum occurrences allowed for this error type. Once a threshold is exceeded, an event is sent to the console specifying the appropriate counter. Additionally, if SNMP traps are enabled, the unit will send a trap with the same information as the console event. |

## Default Values

The default values for this command are as follows:

thresholds BES 15Min 10
thresholds BES 24Hr 100
thresholds CSS 15Min 1
thresholds CSS 24Hr 4
thresholds DM 15Min 1
thresholds DM 24Hr 4
thresholds ES 15Min 65
thresholds ES 24Hr 648
thresholds LCV 15Min 13340
thresholds LCV 24Hr 133400
thresholds LES 15Min 65
thresholds LES 24Hr 648
thresholds PCV 15Min 72
thresholds PCV 24Hr 691
thresholds SES 15Min 10
thresholds SES 24Hr 100
thresholds SEFS 15Min 2
thresholds SEFS 24Hr 17
thresholds UAS 15Min 10
thresholds UAS 24Hr 10

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example sets the threshold for the 15 minute and 24 hour bursty errored seconds counter to **25** and **200**, respectively:

(config)**#thresholds BES 15Min 25**
(config)**#thresholds BES 24Hr 200**

# timing-source

Use the **timing-source** command to configure the timing source used for reference timing. Use the **no** form of this command to return to default settings. Variations of this command include the following:

**timing-source internal**
**timing-source internal secondary**
**timing-source t1** <*interface id*>
**timing-source t1** <*interface id*> **secondary**

## Syntax Description<

| | |
|---|---|
| **internal** | Configures the unit to provide timing using the internal 1.544 MHz clock generator. |
| **t1** <*interface id*> | Configures the unit to recover clocking from the specified T1 or DSX-1 interface. |
| **secondary** | Optional. Signifies that the clock source specified in the command is to be the secondary clock source. |

## Default Values

By default, the primary clock source is set to **t1 0/1**, and the secondary clock source is set to **t1 0/2**.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced to replace the **clock source** command. |

## Functional Notes

If both the primary and secondary clock sources fail, the unit automatically switches to internal timing.

## Usage Examples

The following example configures the unit to use an internal timing source:

(config)**#timing-source internal**

The following examples set the t1 0/1 interface as the primary timing source and the t1 0/2 interface as the secondary timing source:

(config)**#timing-source t1 0/1**
(config)**#timing-source t1 0/2 secondary**

# track *<name>*

Use the **track** commands to create a track as part of network monitoring. This command is also used to enter into the Network Monitoring Track command set once a track is created. These additional commands are covered in *Network Monitor Track Configuration Command Set* on page 1698. Use the **no** form of this command to delete the track.

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies the name of the track being created. |

## Default Values

By default, there are no tracks configured.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Functional Notes

Track objects can be associated with up to two probes to monitor their states. Upon a change in the probe state, the probe sends an event to any track registered with the probe. In response, the track performs the action indicated.

Associating track objects with probes can be defined through using logical AND/OR statements. Refer to *test probe* on page 1702 for more information.

## Usage Examples

The following example creates an track called **track_a**:

```
>enable
#configure terminal
(config)#track track_a
(config-track-track_a)#
```

## Technology Review

Tracks are objects created to monitor network probes for a change in their state. The tracks can be configured to perform a specific action based upon the probe state detected. Association between a track and a probe occurs through referencing the probe in the track's configuration. Once the track is registered with the probe, whenever a change occurs with the probe's state, an event is sent to the track.

Additional configuration commands are available for creating probes. These are explained in the *Network Monitor Probe Command Set* on page 1682.

# **username** <*username*> **password** <*password*>

Use the **username password** command to configure the username and password to use for all protocols requiring a user name-based authentication system including FTP server authentication, line (login local-user list), and HTTP access. Use the **no** form of this command to remove a user name and password.

## Syntax Description

| | |
|---|---|
| <*username*> | Specifies a user name using an alphanumerical string up to 30 characters in length (the user name is case-sensitive). |
| <*password*> | Specifies a password using an alphanumerical string up to 30 characters in length (the password is case-sensitive). |

## Default Values

By default, there is no established user name and password.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

All users defined using the **username/password** command are valid for access to the unit using the **login local-userlist** command.

## Usage Examples

The following example creates a user name of **ADTRAN** with password **ADTRAN**:

(config)**#username ADTRAN password ADTRAN**

# vlan <*vlan id*>

Use the **vlan** command to enter the VLAN Configuration mode. Use the **no** form of this command to remove a VLAN ID. Refer to the section *VLAN Configuration Command Set* on page 1529 for more information.

## Syntax Description

| | |
|---|---|
| <*vlan id*> | Specifies a valid VLAN ID. Range is 1 to 4094. |

## Default Values

No defaults necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following example enters the VLAN configuration mode for VLAN 1:

```
(config)#vlan 1
(config-vlan 1)#
```

# voice alias <*name*> equals <*number*>

Use the **voice alias** <*name*> **equals** <*number*> command to configure the name and number for the call routing alias. This feature allows a single call destination to be reached by an alternate name and number. Use the **no** form of the command to disable this feature.

## Syntax Description

| | |
|---|---|
| <*name*> | Specifies the alias name to describe the call destination. |
| <*number*> | Assigns the alias number to mask the original number. |

## Default Values

No default values necessary for this command.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example activates the **Lobby** voice alias at extension 4100:

config**#voice alias Lobby equals 4100**

# voice ani match <*number*> substitute <*number*>

Use the **voice ani match** command to configure the inbound automatic number identification (ANI) substitution parameters. Use the **no** form of this command to delete substitution parameters.

## Syntax Description

| | |
|---|---|
| **match** <*number*> | Specifies the match template for ANI substitution. |
| **substitute** <*number*> | Specifies number to substitute for the match number. |

## Default Values

No default value necessary for this command.

## Functional Notes

The following rules apply to match numbers:

1) All "," characters are ignored.
2) All "[" and "]" brackets must match and contain numbers only.
3) If using a "$" wildcard, it is the only character allowed.
4) "X" matches [0 to 9], "N" matches [2 to 9].

The following rules apply to substitute numbers:

1) All "," characters are ignored.
2) All "[" and "]" brackets must match and contain numbers only.
3) If using a "$" wildcard, it must be at the end of the number.
4) "X" matches [0 to 9], "N" matches [2 to 9].
5) All wildcard positions must have same position as MATCH number.

The following examples show possible match and substitute numbers:

| MATCH # | SUBST # |
|---|---|
| 1) 256-555-XXXX | 9-555-XXXX |
| 2) NXX-NXX-XXXX | 9-1-NXX-NXX-XXXX |
| 3) $ | 9$ |

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example will substitute the number 256-555-6000 for all matches of 256-555-8000:

(config)**#voice ani match 256555800 substitute 2565556000**

# voice autoattendant

Use the **voice autoattendant** command to configure the auto-attendant options for the system. Use the **no** form of the commands to disable the settings. For more voice auto-attendant options, refer to *voice call-appearance-mode* on page 697. Variations of this command include the following:

**voice autoattendant** <*name*>
**voice autoattendant** <*name*> **extension** <*number*>
**voice autoattendant alias** <*name*>
**voice autoattendant did** <*number*>
**voice autoattendant extension** <*number*>

## Syntax Description

| | |
|---|---|
| <*name*> | Specifies an name of this auto-attendant. |
| **alias** <*name*> | Specifies an alias name to use as an alternate when accessing the auto-attendant. |
| **did** <*number*> | Configures the direct inward dial number to assign to the auto-attendant. |
| **extension** <*number*> | Specifies the extension for auto-attendant system login access. |

## Default Values

No default values necessary for this command.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example configures the **aOperator** as an alias for the auto-attendant:

(config)**#voice autoattendant alias aOperator**

# voice call-appearance-mode

Use the **voice call-appearance-mode** command to configure the unit to allow single or multiple call appearances to user account phones. Use the **no** form of the command to return to default settings. Variations of this command include:

**voice call-appearance-mode multiple**
**voice call-appearance-mode single**

## Syntax Description

| | |
|---|---|
| **multiple** | Allows multiple call appearances. For analog phones, this will be limited to two; for SIP phones the number allowed depends on the model of phone. |
| **single** | Allows only a single call appearance. |

## Default Values

By default, this is set to **single**.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Functional Notes

Each incoming call is classified as a “call appearance.” For example, call waiting supports two call appearances simultaneously. Without call waiting, only one call appearance is supported at a time.

## Usage Examples

The following example sets the unit to allow multiple call appearances:

(config)**#voice call-appearance-mode multiple**

# voice cause-code-map

Use the **voice cause-code-map** command to configure the cause code and SIP message numbers for the PRI. Cause codes and SIP message numbers are associated with a particular connection failure and notifies the system when problems occur. Use the **no** form of the command to return to default settings. Variations of this command include:

**voice cause-code-map from-pri** <*value*> <*value*>
**voice cause-code-map to-pri** <*value*> <*value*>

## Syntax Description

| | |
|---|---|
| **from-pri** <*value*> <*value*> | Enter the cause code number to map to the SIP message. The valid range is 1 to 127. Next, enter the SIP message number to be used. The valid range is 400 to 606 |
| **to-pri** <*value*> <*value*> | Enter the SIP message number to map to the PRI cause code map. The valid range is 400 to 606. The second <*value*> is the PRI cause code number. The valid range is 1 to 127. |

## Default Values

No default values necessary for this command.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example sets the cause code number to 28 to associate with SIP messages:

(config)**#voice cause-code-map from-pri 28**

# voice class-of-service <*set name*>

Use the **voice class-of-service** command to create a voice class of service (CoS) rule set and to enter the Voice Class of Service Command set. Use the **no** form of this command to delete a configured class of service rule set.

## Syntax Description

| | |
|---|---|
| <*set name*> | Specifies the name of the class of service rule set. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example creates a new class of service rule set called **set1**:

(config)**#voice class-of-service set1**
Configuring New Level “set1”.
(config-cos-set1)#

# voice codec-list <*name*>

Use the **voice codec-list** command to create a named CODEC list for call negotiation and to enter the CODEC List command set. Refer to *Voice CODEC List Configuration Command Set* on page 1747 for details. Use the **no** form of this command to delete a configured CODEC list.

## Syntax Description

| | |
|---|---|
| <*name*> | Specifies the CODEC list name. |

## Default Values

By default, there are no configured voice CODEC lists.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example creates a new CODEC list named **list 1**:

(config)**#voice codec-list list 1**
Configuring New Codec List “list 1".
(config-codec)

# voice codec-priority

Use the **voice codec-priority** command to specify which CODEC list (**trunk** or **user**) to set as the priority list. Use the **no** form of the command to disable the settings. Variations of this command include:

**voice codec-priority trunk**
**voice codec-priority user**

## Syntax Description

| | |
|---|---|
| **trunk** | Specifies using the trunk's CODEC list as the priority CODEC list. |
| **user** | Specifies using the user's CODEC list as the priority CODEC list. |

## Default Values

No default values necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example specifies using the trunk's CODEC list as the priority CODEC list:

(config)**#voice codec-priority trunk**

# voice country-code *<number>*

Use the **voice country-code** command to enter the country code for this location. The country code is used to determine if an incoming call is international. If a call originates from the SIP WAN and it is matches the **voice country-code** number, the call is not international and the country code is stripped before passing the call the customer premise equipment (CPE). Use the **no** form of this command to delete the country code.

## Syntax Description

| | |
|---|---|
| *<number>* | Specifies the country code of this location (three digit maximum). |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example specifies a **voice country-code** of **203**:

(config)**#voice country-code 203**

# voice coverage *<name>*

Use the **voice coverage** command to create and modify a global call coverage list to be used to control call routing when a user's phone is not answered. Use the **no** form of the command to disable the settings.

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies a name for the call coverage list. |

## Default Values

No default values necessary for this command.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example specifies that the call coverage list named **absent** to be used for global call coverage:

(config)**#voice coverage absent**

# voice dial-plan

Use the **voice dial-plan** command to add a global number complete pattern. Use the **no** form of this command to delete configured dial plans. Variations of this command include:

**voice dial-plan** <*pattern id*> **900-number** <*pattern*>
**voice dial-plan** <*pattern id*> **always-permitted** <*pattern*>
**voice dial-plan** <*pattern id*> **extensions** <*pattern*>
**voice dial-plan** <*pattern id*> **internal-operator** <*pattern*>
**voice dial-plan** <*pattern id*> **international** <*pattern*>
**voice dial-plan** <*pattern id*> **local** <*pattern*>
**voice dial-plan** <*pattern id*> **long-distance** <*pattern*>
**voice dial-plan** <*pattern id*> **operator-assisted** <*pattern*>
**voice dial-plan** <*pattern id*> **specify-carrier** <*pattern*>
**voice dial-plan** <*pattern id*> **toll-free** <*pattern*>
**voice dial-plan** <*pattern id*> **[user1 | user 2 | user3]** <*pattern*>

## Syntax Description

| | |
|---|---|
| <*pattern id*> | Specifies dial pattern identification. Valid range is 1 to 255. |
| **900-number** | Adds a pattern to the 900 number group. |
| **always-permitted** | Adds a pattern to the always permitted group. |
| **extensions** | Adds a pattern to the internal group. |
| **internal-operator** | Adds a pattern to the internal operator group. |
| **international** | Adds a pattern to the international group. |
| **local** | Adds a pattern to the local group. |
| **long-distance** | Adds a pattern to the long distance group. |
| **operator-assisted** | Adds a pattern to the operator assisted group. |
| **specify-carrier** | Adds a pattern to the specify carrier group. |
| **toll-free** | Adds a pattern to the toll free group. |
| **user1** | Adds a pattern to the user 1 group. |
| **user2** | Adds a pattern to the user 2 group. |
| **user3** | Adds a pattern to the user 3 group. |
| <*pattern*> | Specifies a pattern (Wildcards: N=2 to 9, M=1 to 8, X=0 to 9, []). |

## Default Values

By default, no dial plans are configured.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example adds the pattern **8000** to the local group:

(config)**#voice dial-plan 1 local 8000**

# voice did <*number*> extension <*extension*>

Use the **voice did extension** command to add a Direct Inward Dial (DID) number. Use the **no** form of this command to delete a configured translation.

## Syntax Description

| | |
|---|---|
| <*number*> | Specifies the direct inward dial lookup number. |
| <*extension*> | Specifies the target account of the DID translation. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example directs DID number **5558123** to extension **8123**:

(config)**#voice did 5558123 extension 8123**

# voice feature-mode

Use the **voice feature-mode** command to configure control of the voice features. Use the **no** form of this command to return to the default setting. Variations of this command include:

**voice feature-mode local**
**voice feature-mode network**

## Syntax Description

| | |
|---|---|
| **local** | Allows voice features to be handled by local unit. |
| **network** | Allows voice features to be handled by the network. |

## Default Values

By default, the voice feature mode is set to **network**.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example sets the control of the voice features to the local unit:

(config)**#voice feature-mode local**

# voice flashhook mode

Use the **voice flashhook mode** command to determine if flashhook events will be interpreted locally or will be forwarded to the far end. Use the **no** form of this command to return to the default setting. Variations of this command include:

**voice flashhook mode interpreted**
**voice flashhook mode transparent**

## Syntax Description

| | |
|---|---|
| **interpreted** | Allows the local unit to interpret flashhook events. |
| **transparent** | Specifies flashhook events to be transparent to the provider. |

## Default Values

By default, the voice flashhook mode is set to **interpreted**.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example sets the flashhook mode to allow the local unit to interpret flashhook events:

(config)**#voice flashhook interpreted**

# voice flashhook threshold *<min time> <max time>*

Use the **voice flashhook threshold** command to configure the minimum and maximum time the switch hook must be held to be interpreted as a flash. Use the **no** form of this command to return to the default setting.

## Syntax Description

| | |
|---|---|
| *<min time>* | Specifies minimum flashhook time in milliseconds. Valid range is from 300 to 1000 milliseconds. |
| *<max time>* | Specifies maximum flashhook time in milliseconds. Valid range is from 300 to 1000 milliseconds. |

## Default Values

By default, the flashhook threshold times are 300 milliseconds (minimum) and 1000 milliseconds (maximum).

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example configures the flashhook thresholds at a minimum of **400** to a maximum of **900**:

(config)**#voice flashhook threshold 400 900**

# voice grouped-trunk *<trunk id>*

Use the **voice grouped-trunk** command to create a grouped trunk and to enter the Grouped Trunk command set. Refer to *Voice Trunk Group Command Set* on page 1853 for details. Use the **no** form of this command to delete a configured grouped trunk.

## Syntax Description

| | |
|---|---|
| *<trunk id>* | Specifies the name of the trunk group. |

## Default Values

By default, there are no configured grouped trunks.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example creates the trunk group **trunk3**:

```
(config)#voice grouped-trunk trunk3
(config-TRUNK3)#
```

# voice hold-reminder

Use the **voice hold-reminder** command to specify how long a call can be on hold before the hold reminder rings the phone again. Use the **no** form of the command to return to default settings. Variations of this command include:

**voice hold-reminder** <*value*>
**voice hold-reminder** <*value*> <*interval*>

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies how long a call can be on hold before the hold reminder rings the phone again. Range is 5 to 30 seconds. |
| <*interval*> | Optional. Specifies the interval at which all subsequent reminder rings will occur. Range is 10 to 120 seconds. |

## Default Values

The defaults for this command are a 10-second hold time before the first reminder ring with 30-second intervals between subsequent rings.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example sets the first reminder ring to occur after the call has been on hold for **20** seconds (with subsequent reminder rings occurring every **15** seconds until the call is picked up):

(config)**#voice hold-reminder 20 15**

# voice international-prefix

Use the **voice international-prefix** command to configure the international prefix for this unit. Use the **no** form of this command to delete a configured prefix. Variations of this command include:

**voice international-prefix** *<prefix>*
**voice international-prefix abbreviated**

## Syntax Description

| | |
|---|---|
| **abbreviated** | Specifies the international prefix be replace with a plus symbol (**+)** in the SIP header. |
| *<prefix>* | Specifies the up to four digits for the prefix. |

## Default Values

By default, there is no configured international prefix.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example configures **011** as the international prefix:

(config)**#voice international-prefix 011**

# voice mail

Use the **voice mail** command to configure voice mail options for the unit. Use the **no** form of the command to disable the settings. Refer to *voice mail check* on page 714 for additional arguments. Variations of this command include the following:

**voice mail alias** <*name*>
**voice mail asterisk**
**voice mail class-of-service** <*name*>
**voice mail did** <*number*>
**voice mail extension** <*extension*>
**voice mail internal**
**voice mail leave-extension** <*extension*>
**voice mail max-login-attempts** <*number*>

## Syntax Description

| | |
|---|---|
| **alias** <*name*> | Specifies an alias name to use as an alternate when accessing voice mail. |
| **asterisk** | Enables voice mail on an external Asterisk server. |
| **class-of-service** <*name*> | Configures the voice mail class of services. |
| **did** <*number*> | Configures the direct inward dial number to assign to voice mail. |
| **internal** | Enables internal voice mail on the compact flash. |
| **extension** <*extension*> | Specifies the extension users will dial to retrieve their voice mail. |
| **leave-extension** <*extension*> | Specifies the extension users will dial to leave a voice mail without ringing an extension. If a user forwards their phone to this extension, their calls will automatically forward to their voice mailbox. |
| **max-login-attempts** <*number*> | Specifies the maximum number login attempt to voice mail accounts. Range is 0 to 9 attempts. |

## Default Values

No default values necessary for this command.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |
| Release 12.1 | Command was expanded. |

## Usage Examples

The following example specifies extension **7500** for voice mail retrieval:

(config)**#voice mail extension 7500**

# voice mail check

Use the **voice mail check** command to configure user parameters for the voice mail check extension. Use the **no** form of the command to disable the settings. Variations of this command include the following:

**voice mail check alias** <*name*>
**voice mail check sip-identity** <*station*> <*Txx*>
**voice mail check sip-identity** <*station*> <*Txx*> **register**
**voice mail check sip-identity** <*station*> <*Txx*> **register auth-name** <*username*> **password** <*password*>

## Syntax Description

| | |
|---|---|
| **alias** <*name*> | Specifies an alias name to use as an alternate when accessing the check extension. |
| <*station*> <*Txx*> | Specifies the station to be used for SIP trunk (e.g., station extension) and the SIP trunk (Txx, e.g. **T01**) through which you will register to the server. |
| **register** | Registers the user to the server. |
| **register auth-name** <*username*> | Sets the username that will be required as authentication for registration to the SIP server. |
| **password** <*password*> | Sets the password that will be required as authentication for registration to the SIP server. |

## Default Values

No default values necessary for this command.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example configures the **voice mail check sip-identity** to use extension **6000** as its identity on trunk **T04**:

(config)**#voice voice mail check sip-identity 6000 T04**

# voice mail leave

Use the **voice mail check** command to configure user parameters for the voice mail leave extension. Use the **no** form of the command to disable the settings. Variations of this command include the following:

**voice mail leave alias** <*name*>
**voice mail leave sip-identity** <*station*> <*Txx*>
**voice mail leave sip-identity** <*station*> <*Txx*> **register**
**voice mail leave sip-identity** <*station*> <*Txx*> **register auth-name** <*username*> **password** <*password*>

## Syntax Description

| | |
|---|---|
| **alias** <*name*> | Specifies an alias name to use as an alternate when accessing the check extension. |
| <*station*> <*Txx*> | Specifies the station to be used for SIP trunk (e.g., station extension) and the SIP trunk (Txx, e.g. **T01**) through which you will register to the server. |
| **register** | Registers the user to the server. |
| **register auth-name** <*username*> | Sets the username that will be required as authentication for registration to the SIP server. |
| **password** <*password*> | Sets the password that will be required as authentication for registration to the SIP server. |

## Default Values

No default values necessary for this command.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example configures the **voice mail leave sip-identity** to use extension **8000** as its identity on trunk **T06**:

(config)**#voice mail check sip-identity 8000 T06**

# voice mail sip-identity

Use the **voice mail sip-identity** command to configure SIP voice mail options for the unit. Use the **no** form of the command to disable the settings. Variations of this command include the following:

**voice mail sip-identity** <*sip ID*> <*sip trunk*>
**voice mail sip-identity** <*sip ID*> <*sip trunk*> **register**
**voice mail sip-identity** <*sip ID*> <*sip trunk*> **register auth-name** <*username*> **password** <*password*>

## Syntax Description

| | |
|---|---|
| <*sip ID*> <*sip trunk*> | Specifies a number to be used as the SIP ID (e.g., station extension) and the SIP trunk through which you will register to the server. |
| **register** | Registers the user to the server. |
| **register auth-name** <*username*> | Sets the user name that will be required as authentication for registration to the SIP server. |
| **password** <*password*> | Sets the password that will be required as authentication for registration to the SIP server. |

## Default Values

No default values necessary for this command.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |
| Release 12.1 | Command was expanded. |

## Usage Examples

The following example specifies trunk **T02** and extension **5800** for **voice mail sip-identity**:

(config)**#voice mail sip-identity 5800 T02**

# voice num-rings *<number>*

Use the **voice num-rings** command to globally specify the number of times a station will ring before beginning the coverage path that attempts to deliver a call to an available party. This setting can be overridden on a per-user basis using the **num-rings** command in the Voice User command set. Refer to *Voice User Configuration Command Set* on page 1922 for more information. Use the **no** form of the command to return to default settings.

## Syntax Description

| | |
|---|---|
| *<number>* | Specifies the number of times a station can ring with no answer. Range is 0 to 9. Setting to 0 allows unlimited rings. |

## Default Values

The default for this command is 0 (unlimited rings).

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example sets a limit on the number of times a station can ring:

(config)**#voice num-rings 8**

# voice operator-group

Use the **voice operator-group** command to access the Voice Operator Ring Group command mode. Use the **no** form of the command to disable the settings.

## Syntax Description

No subcommands. Refer to *Voice Ring Group Command Set* on page 1811 for more information.

## Default Values

No default values necessary for this command.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example enters the Operator Group configuration mode:

(config)**#voice operator-group**

Configuring Operator Group.

(config-operator-group)#

# voice overhead-paging extension <*number*>

Use the **voice overhead-paging extension** command to specify the extension used for overhead paging. Use the **no** form of the command to disable the settings.

## Syntax Description

| | |
|---|---|
| <*number*> | Specifies the extension to use for overhead paging. |

## Default Values

No default values necessary for this command.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example configures extension **3000** to be used for overhead paging:

(config)**#voice overhead-paging extension 3000**

# voice park-return <*value*>

Use the **voice park-return** command to configure the time until a parked call returns. Use the **no** form of this command to return to the default setting.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies time in seconds until a call returns from park if not retrieved. Valid range is 15 to 360 seconds. |

## Default Values

By default, the voice park return time is set to 60 seconds.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example sets the time a call returns from park to **30** seconds:

(config)**#voice park-return 30**

# voice ring-group <*number*>

Use the **voice ring-group** command to create or modify ring group parameters. Use the **no** form of this command to delete a configured ring group.

## Syntax Description

| | |
|---|---|
| <*number*> | Specifies the ring group's four digit extension. |

## Default Values

By default, no ring groups are configured.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example creates a new ring group with extension **5678**:

(config)**#voice ring-group 5678**
Configuring New Group “5678”.
(config-5678)#

# voice service-mode

Use the **voice service-mode** command to add a service mode transition. Variations of this command include:

**voice service-mode day** *<day> <time>*
**voice service-mode lunch** *<day> <time>*
**voice service-mode night** *<day> <time>*
**voice service-mode weekend** *<day> <time>*

## Syntax Description

| | |
|---|---|
| **day** | Specifies a transition to day mode. |
| **lunch** | Specifies a transition to lunch mode. |
| **night** | Specifies a transition to night mode. |
| **weekend** | Specifies a transition to weekend mode. |
| *<day>* | Specifies the day of week the transition occurs. |
| *<time>* | Specifies the time for transition to occur (24 hour format - HH:MM). |

## Default Values

By default, the voice service mode is set to **day**.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example sets the voice service mode to **day** with a transition day of Monday and a transition time of 8:00 AM:

(config)**#voice service-mode day monday 08:00**

# voice speed-dial *<unique id> <number> <name>*

Use the **voice speed-dial** command to create a list of IDs to be used as shortcuts to contact frequently called numbers. Use the **no** form of the command to return to default settings.

## Syntax Description

| | |
|---|---|
| *<unique id>* | The speed-dial number that will be used to contact the *<number>* specified. |
| *<number>* | Phone number associated with the speed-dial entry (digits only). |
| *<name>* | Description of this speed-dial entry. |

## Default Values

No default values necessary for this command.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example sets a speed-dial number of **8** for extension **9654**:

(config)**#voice speed-dial 8 9654 3rdFloorLab**

# voice spre *<id> <pattern>*

Use the **voice spre** command to add a global Special PREfix (SPRE) complete pattern.This command allows a user to enter a custom SPRE code. Use the **no** form of this command to delete a configured SPRE pattern.

## Syntax Description

| | |
|---|---|
| *<id>* | Specifies the SPRE pattern ID. Valid range is 1 to 255. |
| *<pattern>* | Specifies complete pattern. Patterns begin with * or # (Wildcards: N = 2 to 9, M = 1 to 8, X = 0 to 9, []). A trailing & allows the use of the dial-plan number complete pattern. |

## Default Values

No defaults are necessary for this command.

## Functional Notes

This command allows the user to enter a SPRE code pattern. If the pattern is followed by an **&**, then the dial-plan templates are used to determine when the unit has enough digits to dial the number (for example, **67&**). However, if a dial plan does not exist for a particular code that is needed, then a SPRE code may be entered followed by a independent dial-plan number complete template (for example, ***67NXX-XXXX**).

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following sets the complete pattern for SPRE 1:

(config)**#voice spre 1 *67NXX-XXXX**

# voice spre-mode

Use the **voice spre-mode** command to control whether special prefix (SPRE) codes will be interpreted by the unit locally or forwarded to the network for interpretation. Use the **no** form of this command to return to default settings. Variations of this command include:

**voice spre-mode local**
**voice spre-mode network**

## Syntax Description

| | |
|---|---|
| **local** | Specifies that SPRE codes are interpreted locally by the unit. |
| **network** | Specifies the forwarding of SPRE codes to the network. |

## Default Values

By default, this command is set to forward SPRE codes to the network.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following examples configures the unit to interpret SPRE codes:

(config)**#voice spre-mode local**

# voice timeouts interdigit <*value*>

Use the **voice timeouts interdigit** command to configure time allowed between dialed digits. Use the **no** form of this command to return to the default setting.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies time in seconds allowed between dialed digits. The valid range is 1 to 16 seconds. |

## Default Values

By default, the voice interdigit timeout is set to 4 seconds.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example sets the time allowed between dialed digits to **10** seconds:

(config)**#voice timeouts interdigit 10**

# voice transfer-mode

Use the **voice transfer-mode** command to specify whether transferred calls will be controlled by the unit locally or if the network will control them. Use the **no** form of this command to return to default settings. Variations of this command include:

**voice transfer-mode local**
**voice transfer-mode network**

## Syntax Description

| | |
|---|---|
| **local** | Specifies that call transferring is controlled locally by the unit. |
| **network** | Specifies that call transferring is controlled by the network. |

## Default Values

By default, the network controls call transfers.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following examples configures the unit to handle call transfers:

(config)**#voice transfer-mode local**

# voice transfer-on-hangup

Use the **voice transfer-on-hangup** command to enable this feature. When transferring a call, hanging up initiates the transfer to the destination party. Use the **no** form of the command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, this feature is enabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example enables this feature:

(config)**#voice transfer-on-hangup**

# voice trunk *<trunk id>* type

Use the **voice trunk type** command to define a new trunk for use with a SIP or ISDN interface. Executing this command activates the Voice Trunk Configuration mode for the individual trunk. Refer to *Voice Trunk ISDN Command Set* on page 1859 for information on the commands in that mode. Other trunk types are explained in the section *voice trunk <trunk id> type t1-rbs supervision* on page 730. Use the **no** form of this command to delete a configured voice trunk. Variations of this command include:

**voice trunk type isdn**
**voice trunk type sip**

> *Refer to the **SIP Trunk Configuration Guide** (61210916L1-29.1) and the **Total Access 900 Series ISDN PRI Quick Configuration Guide** (61210916L1-42.5) for more information on voice trunks. These documents are located on the ADTRAN OS System Documentation CD provided with your unit.*

## Syntax Description

| | |
|---|---|
| *<trunk id>* | Specifies the trunk's two-digit identifier in the format Txx (for example, T01). |
| **isdn** | Configures the trunk for use with ISDN service. |
| **sip** | Configures this trunk for use with SIP. |

## Default Values

No default necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |
| Release 11.1 | Command was expanded to include analog and ISDN support. |

## Usage Examples

The following example creates the new trunk T12 for use with SIP and enters the Voice Trunk Configuration mode:

```
(config)#voice trunk t12 type sip
(config-T12)#
```

# voice trunk *<trunk id>* type t1-rbs supervision

Use the **voice trunk type t1-rbs supervision** command to define a new trunk for a T1 interface. Executing this command activates the Voice Trunk T1 Configuration mode for the individual trunk. Use the **no** form of this command to delete a configured voice trunk. Refer to *Voice Trunk T1 Command Set* on page 1895 for information on the commands in that mode. Other trunk types are explained in the section *voice trunk <trunk id> type* on page 729. Variations of this command include the following:

**voice trunk** *<trunk id>*
**voice trunk** *<trunk id>* **type t1-rbs supervision fgd role user**
**voice trunk** *<trunk id>* **type t1-rbs supervision ground-start role user**
**voice trunk** *<trunk id>* **type t1-rbs supervision immediate role network**
**voice trunk** *<trunk id>* **type t1-rbs supervision immediate role user**
**voice trunk** *<trunk id>* **type t1-rbs supervision loop-start role user**
**voice trunk** *<trunk id>* **type t1-rbs supervision tie-fgd**
**voice trunk** *<trunk id>* **type t1-rbs supervision wink role network**
**voice trunk** *<trunk id>* **type t1-rbs supervision wink role user**

## Syntax Description

| | |
|---|---|
| *<trunk id>* | Specifies the trunk's two-digit identifier in the format Txx (for example, T01). |
| **fgd** | Specifies Feature Group D with an assumed user role. |
| **ground-start** | Specifies ground start with an assumed user role. |
| **immediate** | Specifies E&M immediate with an assumed network or user role. |
| **loop-start** | Specifies loop start with an assumed user role. |
| **tie-fgd** | Specifies tie trunk with Feature Group D. |
| **wink** | Specifies wink with an assumed network or user role. |
| **role network** | Specifies the network role for this trunk. |
| **role user** | Specifies the user role for this trunk. |

## Default Values

No default necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |
| Release 12.1 | Command was expanded. |

## Usage Examples

The following example creates the new trunk **T15** for use with a T1 interface and enters the Voice Trunk T1 Wink Configuration mode:

(config)**#voice trunk t15 type t1-rbs supervision wink role network**
(config-T15)#

# voice user *<extension>*

Use the **voice user** command to create a new user extension and to enter the Voice User command set. Use the **no** form of this command to delete a configured extension or modify an existing extension's parameters. Refer to *Voice User Configuration Command Set* on page 1922 for details.

## Syntax Description

| | |
|---|---|
| *<extension>* | Specifies user's extension. |

## Default Values

By default, there are no configured voice users.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example creates a new user with extension **9876**:

(config)#**voice user 9876**
Configuring New User "9876".
(config-9876)#

# LINE (CONSOLE) INTERFACE CONFIG COMMAND SET

To activate the Line (Console) Interface Configuration mode, enter the **line console 0** command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#line console 0
(config-con 0)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:



All other commands for this command set are described in this section in alphabetical order.



# accounting commands

Use the **accounting commands** command to assign AAA accounting methods to lines. You must first turn AAA on for this command to become available. Use the **no** form of this command to disable this feature. Variations of this command include:

**accounting commands** <*level*> <*name*>
**accounting commands** <*level*> **default**

## Syntax Description

| | |
|---|---|
| <*level*> | Specifies a command level. Choose from 1 or 15. |
| <*name*> | Applies a named accounting method to this line. |
| **default** | Applies the default accounting method to a line. |

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example applies the default accounting method to line 1:

(config)**#aaa on**
(config)**#line console 0**
(config-con0)**#accounting commands 1 default**

# accounting connection

Use the **accounting commands** command to assign AAA accounting methods to lines. You must first turn AAA on for this command to become available. For more detailed information on AAA functionality, refer to the *Technology Review* section of the command *aaa on* on page 441. Use the **no** form of this command to disable this feature. Variations of this command include the following:

**accounting connection** *<name>*
**accounting connection default**

## Syntax Description

| | |
|---|---|
| *<name>* | Applies a named accounting method to this line. |
| **default** | Applies the default accounting method to a line. |

## Default Values

By default, there is no accounting method applied to a line.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example specifies using the default accounting method to account the outbound Telnet sessions on this line:

(config)**#aaa on**
(config)**#line console 0**
(config-console0)**#accounting connection default**

# accounting exec

Use the **accounting commands** command to assign AAA accounting methods to lines. You must first turn AAA on for this command to become available. For more detailed information on AAA functionality, refer to the *Technology Review* section of the command *aaa on* on page 441. Use the **no** form of this command to disable this feature. Variations of this command include the following:

**accounting exec** *<name>*
**accounting exec default**

## Syntax Description

| | |
|---|---|
| *<name>* | Applies a named accounting method to this line. |
| **default** | Applies the default accounting method to a line. |

## Default Values

By default, there is no accounting method applied to a line.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example specifies using the default accounting method to account all inbound Telnet sessions on this line:

(config)**#aaa on**
(config)**#line console 0**
(config-console0)**#accounting exec default**

# authorization commands

Use the **authorization commands** command to assign AAA authorization methods to lines. You must first turn AAA on for this command to become available. Use the **no** form of this command to disable this feature. Variations of this command include:

**authorization commands** <*level*> <*name*>
**authorization commands** <*level*> **default**

## Syntax Description

| | |
|---|---|
| <*level*> | Specifies a command level. Choose from 1 or 15. |
| <*name*> | Applies a named authorization method to this line. |
| **default** | Applies the default authorization method to a line. |

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example applies the default authorization method to line 1:

(config)**#aaa on**
(config)**#line console 0**
(config-con0)**#authorization commands 1 default**

# authorization exec

Use the **authorization exec** command to assign AAA authorization methods to lines. You must first turn AAA on for this command to become available. For more detailed information on AAA functionality, refer to the *Technology Review* section of the command *aaa on* on page 441. Use the **no** form of this command to disable this feature. Variations of this command include the following:

**authorization exec** *<name>*
**authorization exec default**

## Syntax Description

| | |
|---|---|
| *<name>* | Applies a named authorization method to this line. |
| **default** | Applies the default authorization method to a line. |

## Default Values

By default, there is no authorization method applied to a line.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example specifies using the default authorization method on this line to authorize an exec session:

(config)**#aaa on**
(config)**#line console 0**
(config-console0))**#authorization exec default**

# **databits** *<value>*

Use the **databits** command to set the number of databits per character for a terminal session. This value must match the configuration of your VT100 terminal or terminal emulator software. The default is 8 databits per character. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| *<value>* | Specifies the data bits per character. Select from 7 or 8 databits per character. |

## Default Values

By default, the databits are set to 8.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example configures **7** databits per character for the console terminal session:

(config)**#line console 0**
(config-con 0)**#databits 7**

# flowcontrol

Use the **flowcontrol** command to set flow control for the line console. Use the **no** form of this command to return to the default setting. Variations of this command include:

**flowcontrol none**
**flowcontrol software in**

## Syntax Description

| | |
|---|---|
| **none** | Specifies no flow control. |
| **software in** | Configures AOS to derive flow control from the attached device. |

## Default Values

By default, flow control is set to **none**.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Usage Examples

The following example configures no flow control for the line console:

(config)**#line console 0**
(config-con 0)**#flowcontrol none**

# line-timeout <*value*>

Use the **line-timeout** command to specify the number of minutes a line session may remain inactive before the AOS terminates the session. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the number of minutes a line session may remain inactive before the AOS terminates the session. Range is 0 to 35,791 minutes.<br>Entering a **line-timeout** value of 0 disables the feature. |

## Default Values

By default the **line-timeout** is set to 15 minutes (Console and Telnet).

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example specifies a timeout of **2** minutes:

(config)**#line console 0**
(config-con 0)**#line-timeout 2**

# login

Use the **login** command to enable security login on the line session requiring the password configured using the **password** command. Use the **no** form of this command to disable the login feature.

## Syntax Description

No subcommands.

## Default Values

By default, there is no login password set for access to the unit.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example enables the security login feature and specifies a password (**mypassword**) on the available console session:

(config)**#line console 0**
(config-console 0)**#login**
(config-console 0)**#password mypassword**

# login authentication *<aaa login list>*

Use the **login authentication** command to specify the named AAA login list to use for authenticating users connecting on this line. Use the **no** form of this command to return to the default setting.

## Syntax Description

| | |
|---|---|
| *<aaa login list>* | Specifies the AAA login list to use for authentication. |

## Default Values

The default value is the default AAA list.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

If the AAA subsystem is activated but no login authentication list is given, the default list is used. If the default list is used but the default list is not configured, the behavior for consoles is to be granted access. This prevents a lockout configuration.

## Usage Examples

The following example specifies that **myList** will be used for authenticating users connecting on this line:

(config)**#line console 0**
(config-con 0)**#login authentication myList**

# login local-userlist

Use the **login local-userlist** command to enable security login for the terminal session requiring the usernames and passwords configured using the **username/password** Global Configuration command. Use the **no** form of this command to disable the login local-userlist feature.

*All user properties assigned using the **username/password** command are valid when using the **login local-userlist** command.*

## Syntax Description

No subcommands.

## Default Values

By default, there is no login password set for access to the unit.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example displays creating a local userlist and enabling the security login feature on the **CONSOLE** port:

(config)**#username my_user password my_password**
(config)**#line console 0**
(config-con 0)**#login local-userlist**

When connecting to the unit, the following prompts are displayed:

User Access Login
Username: **ADTRAN**
Password:
Router#

# parity

Use the **parity** command to specify the type of parity used as error correction. This value must match the configuration of your VT100 terminal or terminal emulator software. Use the **no** form of this command to return to the default value. Variations of this command include:

**parity even**
**parity mark**
**parity none**
**parity odd**
**parity space**

## Syntax Description

| | |
|---|---|
| **even** | Sets the parity bit to 0 if the number of 1 bits in the data sequence is odd, or set to 1 if the number of 1 bits is even. |
| **mark** | Always sets the parity bit to 1. |
| **none** | No parity bit used. |
| **odd** | Sets the parity bit to 1 if the number of 1 bits in the data sequence is even, or set to 1 if the number is odd. |
| **space** | Always sets the parity bit to 0. |

## Default Values

By default, the parity option is set to **none**.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

Parity is the process used to detect whether characters have been altered during the data transmission process. Parity bits are appended to data frames to ensure that parity (whether it be odd or even) is maintained.

## Usage Examples

The following example specifies **mark** parity for the console terminal session:

(config)**#line console 0**
(config-con 0)**#parity mark**

# password *<password>*

Use the **password** command to configure the password (with optional encryption) required on the line session when security login is enabled (using the **login** command). Use the **no** form of this command to remove a configured password. Variations of this command include:

**password** *<password>*
**password md5** *<password>*

## Syntax Description

| | |
|---|---|
| *<password>* | Specifies the password for the line session using an alphanumeric character string (up to 16 characters). |
| **md5** | Specifies Message Digest 5 (MD5) as the encryption protocol to use when displaying the enable password during show commands. If the MD5 keyword is not used, encryption is not used when displaying the enable password during show commands. |

## Default Values

By default, there is no login password set for access to the unit.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 6.1 | Encryption was added. |

## Usage Examples

The following example enables the security login feature and specifies a password on the **CONSOLE** port:

(config)**#line console 0**
(config-con 0)**#login**
(config-con 0)**#password mypassword**

To provide extra security, the AOS can encrypt the enable password when displaying the current configuration. For example, the following is a **show configuration** printout (password portion) with an unencrypted enable password (ADTRAN):

!
enable password ADTRAN
!

Alternately, the following is a **show configuration** printout (password portion) with an enable password of ADTRAN using md5 encryption:

!
enable password md5 encrypted 5aa5fbae7d01a90e79fb57705ce74676
!

# **speed** *<rate>*

Use the **speed** command to specify the data rate for the **CONSOLE** port. This setting must match your VT100 terminal emulator or emulator software. Use the **no** form of this command to restore the default value.

## Syntax Description

| | |
|---|---|
| *<rate>* | Specifies rate of data transfer on the interface (2400; 4800; 9600; 19,200; 38,400; 57,600; or 115,200 bps). |

## Default Values

By default, the speed is set to 9600 bps.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example configures the **CONSOLE** port for 19,200 bps:

(config)**#line console 0**
(config-con 0)**#speed 19200**

# **stopbits** *<value>*

Use the **stopbits** command to set the number of stopbits per character for a terminal session. This value must match the configuration of your VT100 terminal or terminal emulator software. The default is 1 stopbit per character. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| *<value>* | Specifies the stopbits per character. Select from 1 or 2 stopbits per character. |

## Default Values

By default, the **stopbits** are set to **1**.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example configures **2** stopbits per character for the console terminal session:

(config)**#line console 0**
(config-con 0)**#stopbits 2**

# LINE (SSH) INTERFACE CONFIG COMMAND SET

To activate the Line Secure Shell (SSH) Interface Configuration mode, enter the **line ssh** command specifying a SSH session(s) at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#line ssh 0 4
(config-ssh0-4)#
```

You can select a single line by entering the **line ssh** command followed by the line number (0-4). For example:

```
>enable
#configure terminal
(config)#line ssh 2
(config-ssh2)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:

*cross-connect on page 29*
*do on page 33*
*end on page 34*
*exit on page 35*
*shutdown on page 36*

All other commands for this command set are described in this section in alphabetical order.

*access-class <name> in on page 749*
*accounting commands begin on page 750*
*authorization commands begin on page 753*
*line-timeout <value> on page 755*
*login on page 756*
*login authentication <aaa login list> on page 757*
*login local-userlist on page 758*

# access-class *<name>* in

Use the **access-class in** command to restrict Secure Shell (SSH) access using a configured access control list. Received packets passed by the access control list will be allowed. Use the access control list configuration to deny hosts or entire networks or to permit specified IP addresses. Use the **no** form of this command to disable this feature. Refer to *ip access-list standard <name>* on page 496 and *ip access-list extended <name>* on page 492 for more information about configuring access control lists.

## Syntax Description

| | |
|---|---|
| *<name>* | Identifies the configured access control list using an alphanumeric descriptor (all access control list descriptors are case-sensitive). |

## Default Values

By default, there are no configured access control lists associated with SH sessions.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Functional Notes

When using the **access-class in** command to associate an access control list with an SSH session, remember to duplicate the **access-class in** command for all configured SSH sessions 0 through 4. SSH access to the unit using a particular SSH session is not possible. Users will be assigned the first available SSH session.

## Usage Examples

The following example associates the access control list **Trusted** (to allow SSH sessions from the 192.22.56.0/24 network) with all SSH sessions (0 through 4):

Create the access control list:

(config)**#ip access-list standard Trusted**
(config)**#permit 192.22.56.0 0.0.0.255**

Enter the line (ssh):

(config)**#line ssh 0 4**

Associate the access control list with the SSH session:

(config-ssh0-4)**#access-class Trusted in**

# accounting commands

Use the **accounting commands** command to assign AAA accounting methods to lines. You must first turn AAA on for this command to become available. Use the **no** form of this command to disable this feature. Variations of this command include:

**accounting commands** <*level*> <*name*>
**accounting commands** <*level*> **default**

## Syntax Description

| | |
|---|---|
| <*level*> | Specifies a command level. Choose from 1 or 15. |
| <*name*> | Applies a named accounting method to this line. |
| **default** | Applies the default accounting method to a line. |

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example applies the default accounting method to line 1:

(config)**#aaa on**
(config)**#line ssh 1**
(config-ssh1)**#accounting commands 1 default**

# accounting connection

Use the **accounting commands** command to assign AAA accounting methods to lines. You must first turn AAA on for this command to become available. For more detailed information on AAA functionality, refer to the *Technology Review* section of the command *aaa on* on page 441. Use the **no** form of this command to disable this feature. Variations of this command include the following:

**accounting connection** *<name>*
**accounting connection default**

## Syntax Description

| | |
|---|---|
| *<name>* | Applies a named accounting method to this line. |
| **default** | Applies the default accounting method to a line. |

## Default Values

By default, there is no accounting method applied to a line.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example specifies using the default accounting method to account the outbound Telnet sessions on this line:

(config)**#aaa on**
(config)**#line ssh 1**
(config-ssh1)**#accounting connection default**

# accounting exec

Use the **accounting commands** command to assign AAA accounting methods to lines. You must first turn AAA on for this command to become available. For more detailed information on AAA functionality, refer to the *Technology Review* section of the command *aaa on* on page 441. Use the **no** form of this command to disable this feature. Variations of this command include the following:

**accounting exec** *<name>*
**accounting exec default**

## Syntax Description

| | |
|---|---|
| *<name>* | Applies a named accounting method to this line. |
| **default** | Applies the default accounting method to a line. |

## Default Values

By default, there is no accounting method applied to a line.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example specifies using the default accounting method to account all inbound Telnet sessions on this line:

(config)**#aaa on**
(config)**#line ssh 1**
(config-ssh1)**#accounting exec default**

# authorization commands

Use the **authorization commands** command to assign AAA authorization methods to lines. You must first turn AAA on for this command to become available. Use the **no** form of this command to disable this feature. Variations of this command include:

**authorization commands** <*level*> <*name*>
**authorization commands** <*level*> **default**

## Syntax Description

| | |
|---|---|
| <*level*> | Specifies a command level. Choose from 1 or 15. |
| <*name*> | Applies a named authorization method to this line. |
| **default** | Applies the default authorization method to a line. |

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example applies the default authorization method to line 1:

(config)**#aaa on**
(config)**#line ssh 1**
(config-ssh1)**#authorization commands 1 default**

# authorization exec

Use the **authorization exec** command to assign AAA authorization methods to lines. You must first turn AAA on for this command to become available. For more detailed information on AAA functionality, refer to the *Technology Review* section of the command *aaa on* on page 441. Use the **no** form of this command to disable this feature. Variations of this command include the following:

**authorization exec** *<name>*
**authorization exec default**

## Syntax Description

| | |
|---|---|
| *<name>* | Applies a named authorization method to this line. |
| **default** | Applies the default authorization method to a line. |

## Default Values

By default, there is no authorization method applied to a line.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example specifies using the default authorization method on this line to authorize an exec session:

(config)**#aaa on**
(config)**#line ssh 1**
(config-ssh1)**#authorization exec default**

# line-timeout <*value*>

Use the *line-timeout* command to specify the number of minutes a line session may remain inactive before the AOS terminates the session. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the number of minutes a line session may remain inactive before the AOS terminates the session. Valid range: 0 to 35,791.<br>Entering a **line-timeout** value of 0 disables the feature. |

## Default Values

By default the **line-timeout** is set to 15 minutes.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example specifies a timeout of 2 minutes for all SSH sessions:

(config)**#line ssh 0 4**
(config-ssh0-4)**#line-timeout 2**

# login

Use the **login** command to enable security login on the line session requiring the password configured using the **password** command. Use the **no** form of this command to disable the login feature.

## Syntax Description

No subcommands.

## Default Values

By default, there is no login password set for access to the unit.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example enables the security login feature and specifies a password on all the available SSH sessions (0 through 4):

(config)**#line ssh 0 4**
(config-ssh0-4)**#login**
(config-ssh0-4)**#password mypassword**

# login authentication *<aaa login list>*

Use the **login authentication** command to assign the named AAA login list to use for authenticating users connecting on this line. Use the **no** form of the command to remove the AAA authentication list.

## Syntax Description

| | |
|---|---|
| *<aaa login list>* | Specifies the name of the AAA login list to use for authentication. |

## Default Values

The default value is the default AAA list.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Functional Notes

If the AAA subsystem is activated but no login authentication list is given, the default list is used. If the default list is used but the default list is not configured, SSH uses the local user database.

## Usage Examples

The following example specifies that **myList** will be used for authenticating users connecting on this line:

(config)**#line ssh 2**
(config-ssh2)**#login authentication myList**

# login local-userlist

Use the **login local-userlist** command to check the local list of usernames and passwords configured using the **username/password** Global Configuration command (refer to *username <username> password <password>* on page 692).

> NOTE *All user properties assigned using the **username/password** command are valid when using the **login local-userlist** command.*

## Syntax Description

No subcommands.

## Default Values

By default, there is no login password set for access to the unit.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example creates a local userlist and enables the security login feature:

(config)**#username my_user password my_password**
(config)**#line ssh 0**
(config-ssh0)**#login local-userlist**

When connecting to the unit, the following prompts are displayed:

User Access Login
Username: **my_user**
Password:
#

# LINE (TELNET) INTERFACE CONFIG COMMAND SET

To activate the Line (Telnet) Interface Configuration mode, enter the **line telnet** command specifying a Telnet session(s) at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#line telnet 0 4
(config-telnet0-4)#
```

You can select a single line by entering the **line telnet** command followed by the line number (0-4). For example:

```
>enable
#configure terminal
(config)#line telnet 2
(config-telnet2)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:



All other commands for this command set are described in this section in alphabetical order.



# access-class *<name>* in

Use the **access-class in** command to restrict Telnet access using a configured access control list. Received packets passed by the access control list will be allowed. Use the access control list configuration to deny hosts or entire networks or to permit specified IP addresses. Use the **no** form of this command to remove a configured access class. Refer to *ip access-list standard <name>* on page 496 and *ip access-list extended <name>* on page 492 for more information about configuring access control lists.

## Syntax Description

| | |
|---|---|
| *<name>* | Identifies the configured access control list using an alphanumeric descriptor (all access control list descriptors are case-sensitive). |

## Default Values

By default, there are no configured access control lists associated with Telnet sessions.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

When using the **access-class in** command to associate an access control list with a Telnet session, remember to duplicate the **access-class in** command for all configured Telnet sessions 0 through 4. Telnet access to the unit using a particular Telnet session is not possible. Users will be assigned the first available Telnet session.

## Usage Examples

The following example associates the access control list **Trusted** (to allow Telnet sessions from the 192.22.56.0/24 network) with all Telnet sessions (0 through 4):

Create the access control list:

(config)**#ip access-list standard Trusted**
(config)**#permit 192.22.56.0 0.0.0.255**

Enter the line (telnet):

(config)**#line telnet 0 4**

Associate the access control list with the Telnet session:

(config-telnet0-4)**#access-class Trusted in**

# accounting commands

Use the **accounting commands** command to assign AAA accounting methods to lines. You must first turn AAA on for this command to become available. Use the **no** form of this command to disable this feature. Variations of this command include:

**accounting commands** <*level*> <*name*>
**accounting commands** <*level*> **default**

## Syntax Description

| | |
|---|---|
| <*level*> | Specifies a command level. Choose from 1 or 15. |
| <*name*> | Applies a named accounting method to this line. |
| **default** | Applies the default accounting method to a line. |

## Default Values

By default, for this accounting commands is disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example applies the default accounting method to Telnet session 1:

(config)**#aaa on**
(config)**#line telnet 1**
(config-telnet1)**#accounting commands 1 default**

# accounting connection

Use the **accounting commands** command to assign AAA accounting methods to lines. You must first turn AAA on for this command to become available. For more detailed information on AAA functionality, refer to the *Technology Review* section of the command *aaa on* on page 441. Use the **no** form of this command to disable this feature. Variations of this command include the following:

**accounting connection** *<name>*
**accounting connection default**

## Syntax Description

| | |
|---|---|
| *<name>* | Applies a named accounting method to this line. |
| **default** | Applies the default accounting method to a line. |

## Default Values

By default, there is no accounting method applied to a line.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example specifies using the default accounting method to account the outbound Telnet sessions on this line:

(config)**#aaa on**
(config)**#line telnet 1**
(config-telnet1)**#accounting connection default**

# accounting exec

Use the **accounting commands** command to assign AAA accounting methods to lines. You must first turn AAA on for this command to become available. For more detailed information on AAA functionality, refer to the *Technology Review* section of the command *aaa on* on page 441. Use the **no** form of this command to disable this feature. Variations of this command include the following:

**accounting exec** *<name>*
**accounting exec default**

## Syntax Description

| | |
|---|---|
| *<name>* | Applies a named accounting method to this line. |
| **default** | Applies the default accounting method to a line. |

## Default Values

By default, there is no accounting method applied to a line.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example specifies using the default accounting method to account all inbound Telnet sessions on this line:

(config)**#aaa on**
(config)**#line telnet 1**
(config-telnet1)**#accounting exec default**

# authorization commands

Use the **authorization commands** command to assign AAA authorization methods to lines. You must first turn AAA on for this command to become available. Use the **no** form of this command to disable this feature. Variations of this command include:

**authorization commands** <*level*> <*name*>
**authorization commands** <*level*> **default**

## Syntax Description

| | |
|---|---|
| <*level*> | Specifies a command level. Choose from 1 or 15. |
| <*name*> | Applies a named authorization method to this line. |
| **default** | Applies the default authorization method to a line. |

## Default Values

By default, authorization commands is disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example applies the default authorization method to line 1:

(config)**#aaa on**
(config)**#line telnet 1**
(config-telnet1)**#authorization commands 1 default**

# authorization exec

Use the **authorization exec** command to assign AAA authorization methods to lines. You must first turn AAA on for this command to become available. For more detailed information on AAA functionality, refer to the *Technology Review* section of the command *aaa on* on page 441. Use the **no** form of this command to disable this feature. Variations of this command include the following:

**authorization exec** *<name>*
**authorization exec default**

## Syntax Description

| | |
|---|---|
| *<name>* | Applies a named authorization method to this line. |
| **default** | Applies the default authorization method to a line. |

## Default Values

By default, there is no authorization method applied to a line.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example specifies using the default authorization method on this line to authorize an exec session:

(config)**#aaa on**
(config)**#line telnet 1**
(config-telnet1)**#authorization exec default**

# line-timeout <*value*>

Use the **line-timeout** command to specify the number of minutes a line session may remain inactive before the AOS terminates the session. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the number of minutes a line session may remain inactive before the AOS terminates the session. Range is 0 to 35,791 minutes.<br>Entering a **line-timeout** value of 0 disables the feature. |

## Default Values

By default the **line-timeout** is set to 15 minutes (Console and Telnet).

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example specifies a timeout of 2 minutes:

(config)**#line telnet 0**
(config-telnet0)**#line-timeout 2**

# login

Use the **login** command to enable security login on the line session requiring the password configured using the **password** command. Use the **no** form of this command to disable the login feature.

## Syntax Description

No subcommands.

## Default Values

By default, there is no login password set for access to the unit.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example enables the security login feature and specifies a password on all the available Telnet sessions (0 through 4):

(config)**#line telnet 0 4**
(config-telnet0-4)**#login**
(config-telnet0-4)**#password mypassword**

# login authentication *<aaa login list>*

Use the **login authentication** command to specify the named AAA login list to use for authenticating users connecting on this line. Use the **no** form of this command to return to the default setting.

## Syntax Description

| | |
|---|---|
| *<aaa login list>* | Specifies the AAA login list to use for authentication. |

## Default Values

The default value is the default AAA list.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

If the AAA subsystem is activated but no login authentication list is given, the default list is used. If the default list is used but the default list is not configured, the behavior for telnets is to use the local user database.

## Usage Examples

The following example specifies that **myList** will be used for authenticating users connecting on this line:

(config)**#line telnet 2**
(config-telnet2)**#login authentication myList**

# login local-userlist

Use the **login local-userlist** command to enable security login for the terminal session requiring the usernames and passwords configured using the **username/password** Global Configuration command. Use the **no** form of this command to disable the login local-userlist feature.

*All user properties assigned using the **username/password** command are valid when using the **login local-userlist** command.*

## Syntax Description

No subcommands.

## Default Values

By default, there is no login password set for access to the unit.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example displays creating a local userlist and enabling the security login feature:

(config)**#username my_user password my_password**
(config)**#line telnet 0**
(config-telnet0)**#login local-userlist**

When connecting to the unit, the following prompts are displayed:

User Access Login
Username: **my_user**
Password:
Router#

# password *<password>*

Use the **password** command to configure the password (with optional encryption) required on the line session when security login is enabled (using the **login** command). Use the **no** form of this command to remove a configured password. Variations of this command include:

**password** *<password>*
**password md5** *<password>*

## Syntax Description

| | |
|---|---|
| *<password>* | Specifies the password for the line session using an alphanumeric character string (up to 16 characters). |
| **md5** | Optional. Specifies Message Digest 5 (MD5) as the encryption protocol to use when displaying the enable password during show commands. If the MD5 keyword is not used, encryption is not used when displaying the enable password during show commands. |

## Default Values

By default, there is no login password set for access to the unit.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example enables the security login feature and specifies a password for the Telnet session 0:

(config)**#line telnet 0**
(config-telnet0)**#login**
(config-telnet0)**#password mypassword**

To provide extra security, the AOS can encrypt the enable password when displaying the current configuration. For example, the following is a **show configuration** printout (password portion) with an unencrypted enable password (ADTRAN):

```
!
enable password ADTRAN
!
```

Alternately, the following is a **show configuration** printout (password portion) with an enable password of ADTRAN using md5 encryption:

```
!
enable password md5 encrypted 5aa5fbae7d01a90e79fb57705ce74676
!
```

# ADSL INTERFACE CONFIGURATION COMMAND SET

To activate the ADSL Interface Configuration mode, enter the **interface adsl** command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#interface adsl 0/1
(config-adsl 0/1)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:



All other commands for this command set are described in this section in alphabetical order.



# retrain

Use the **retrain** command to force the modem to retrain.

## Syntax Description

No subcommands.

## Default Values

No default is necessary for this command.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Usage Examples

The following example forces a modem retrain:

```
(config)#interface adsl 0/1
(config-adsl 0/1)#retrain
```

# snr-margin

Use the **snr-margin** command to enable monitoring and set the minimum signal-to-noise ratio (SNR) during training and showtime. Use the **no** form of this command to disable monitoring. Variations of this command include:

**snr-margin** <*margin*>
**snr-margin showtime monitor**
**snr-margin training monitor**

## Syntax Description

| | |
|---|---|
| <*margin*> | Sets the minimum SNR margin value in dB. The range is from 1 to 15 dB. |
| **showtime monitor** | Enables margin monitoring to retrain the ADSL interface if the specified minimum margin is violated during showtime. |
| **training monitor** | Enables margin monitoring to retrain the ADSL interface if the specified minimum margin is violated during training. |

## Default Values

By default, SNR margin monitoring is disabled.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Usage Examples

The following example enables SNR margin monitoring during showtime with a minimum level of 7 dB:

(config)**#interface adsl 0/1**
(config-adsl 0/1)**#snr-margin showtime monitor 7**

# training-mode

Use the **training-mode** command to configure the ADSL training mode. Use the **no** form of these commands to disable a specific training mode. Variations of this command include:

**training-mode ADSL2**
**training-mode ADSL2+**
**training-mode ADSL2+ANNEX-M**
**training-mode G.DMT**
**training-mode G.LITE**
**training-mode Multi-Mode**
**training-mode READSL2**
**training-mode T1.413**

## Syntax Description

| | |
|---|---|
| **ADSL2** | Specifies ITU G.992.3 Annex A mode. |
| **ADSL2+** | Specifies ITU G.992.5 ADSL2+ mode. |
| **ADSL2+ANNEX-M** | Specifies ITU G.992.5 Annex M ADSL2+ mode. |
| **G.DMT** | Specifies ANSI full-rate mode. |
| **G.LITE** | Specifies ANSI splitterless mode. |
| **Multi-Mode** | Specifies auto detect mode. When set to multi-mode, the ADSL interface attempts to train to the DSLAM using each of the supported training modes until a match is found. |
| **READLS2** | Specifies ITU G.992.3 Annex L mode. |
| **T1.413** | Specifies ANSI T1.413 mode. |

## Default Values

By default, the training mode is set to **Multi-Mode**.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |
| Release 13.1 | Command was expanded to include ITU G.992.5 Annex M ADSL2+ mode. |

## Functional Notes

Some of the listed training modes (G.LITE, T1.413, ADSL2, ADSL2+, READSL2) are currently supported for ADSL over POTS (Annex A) and are not valid for ADSL over ISDN (Annex B) modules.

## Usage Examples

The following example sets the training mode to **T1.413**:

(config)**#interface adsl 0/1**
(config-adsl 0/1)**#training-mode T1.413**

# BRI INTERFACE CONFIGURATION COMMAND SET

To activate the BRI Interface Configuration mode, enter the **interface bri** command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#interface bri 1/2
(config-bri 1/2)#
```

*The BRI interface number in the example above is shown as **bri 1/2**. This number is based on the interface's location (slot/port) and could vary depending on the unit's configuration. Use the **do show interfaces** command to determine the appropriate interface number.*

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:



All other commands for this command set are described in this section in alphabetical order.



# caller-id override

Use the **caller-id override** command to configure the unit to replace caller ID information with a user-specified number. Use the **no** form of this command to disable any caller ID overrides. Variations of this command include:

**caller-id override always** <*number*>
**caller-id override if-no-cid** <*number*>

## Syntax Description

| | |
|---|---|
| **always** <*number*> | Always forces replacement of the incoming caller ID number with the number given. |
| **if-no-cid** <*number*> | Replaces the incoming caller ID number with the number given only if there is no caller ID information available for the incoming call. |

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Functional Notes

This command forces a replacement of the incoming caller ID number with the number given. The received caller ID, if any, is discarded, and the given override number is used to connect the incoming call to a circuit of the same number.

## Usage Examples

The following example configures the unit to always provide the given number as the caller ID number:

(config)**#interface bri 1/2**
(config-bri 1/2)**#caller-id override always 5551000**

# isdn ldn

Use the **isdn ldn** command to specify the Local Directory Numbers (LDNs) for the Basic Rate ISDN (BRI) interface. This information should be supplied by your service provider. Use the **no** form of this command to remove a configured LDN. Variations of this command include:

**isdn ldn1** *<ldn number>*
**isdn ldn2** *<ldn number>*

*The BRI module requires all incoming calls to be directed to the Local Directory Number (LDN) associated with the SPID programmed using the **isdn spid1** command. All calls to the LDN associated with SPID 2 will be rejected (unless part of a bonding call).*

## Syntax Description

| | |
|---|---|
| **ldn1** | Specifies the LDN associated with the SPID entered as **spid1**. |
| **ldn2** | Specifies the LDN associated with the SPID entered as **spid2**. |
| *<ldn number>* | Specifies the LDN assigned to the circuit by the service provider. The LDN is the number used by remote callers to dial into the ISDN circuit. |

## Default Values

By default, there are no configured LDNs.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Functional Notes

Inbound calls are not accepted on interfaces without programmed LDNs. LDNs can also be entered using the **isdn spid** command. The **isdn spid** and **isdn ldn** commands overwrite the existing programmed LDN; therefore, the last LDN programming entered takes precedence.

## Usage Examples

The following example defines an LDN of 555-1111:

(config)**#interface bri 1/2**
(config-bri 1/2)**#isdn ldn1 5551111**

# isdn spid

Use the **isdn spid** command to specify the Service Profile Identifiers (SPIDs) and the Local Directory Numbers (LDNs) for the Basic Rate ISDN (BRI) interface. This information should be supplied by your service provider. Use the **no** form of this command to remove a configured SPID. Variations of this command include:

**isdn spid1** <*spid number*> <*ldn number*>
**isdn spid2** <*spid number*> <*ldn number*>

*The BRI module requires all incoming calls to be directed to the Local Directory Number (LDN) associated with the SPID programmed using the **isdn spid1** command. All calls to the LDN associated with SPID 2 will be rejected (unless part of a bonding call).*

## Syntax Description

| | |
|---|---|
| **spid1** | Specifies the primary SPID. |
| **spid2** | Specifies the secondary SPID. |
| <*spid number*> | Specifies the 8 to 14 digit number identifying your BRI line in the central office switch. A SPID is generally created using the area code and phone number associated with the line and a four-digit suffix. For example, the following SPIDs may be provided on a BRI line with phone numbers 555-1111 and 555-1112:<br>SPID 1: 701 555 1111 0101<br>SPID 2: 701 555 1112 0101 |
| <*ldn number*> | Optional. Specifies the LDN assigned to the circuit by the service provider. An LDN programmed using the **isdn spid1** command is automatically associated with SPID 1. An LDN programmed using the **isdn spid2** command is automatically associated with SPID 2. The LDN is the number used by remote callers to dial into the ISDN circuit. Inbound calls are not accepted on interfaces without programmed LDNs. LDNs can also be entered using the **isdn ldn** command. The **isdn spid** and **isdn ldn** commands overwrite the existing programmed LDN; therefore the last LDN programming entered takes precedence. |

## Default Values

By default, there are no configured SPIDs or LDNs.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

The AOS does not support “SPID-less” 5ESS signaling. SPIDs are required for all configured BRI endpoints using 5ESS signaling.

For European applications, a SPID is not necessary. Use the **isdn ldn** command to configure the LDN for European applications.

## Usage Examples

The following example defines a SPID of 704 555 1111 0101 with an LDN of 555 1111:

(config)**#interface bri 1/2**
(config-bri 1/2)**#isdn spid1 70455511110101 5551111**

# isdn switch-type

Use the **isdn switch-type** command to specify the ISDN signaling type configured on the Basic Rate ISDN (BRI) interface. The type of ISDN signaling implemented on the BRI interface does not always match the manufacturer of the Central Office switch. Use the **no** form of this command to return to the default value. Variations of this command include:

**isdn switch-type basic-5ess**
**isdn switch-type basic-dms**
**isdn switch-type basic-net3**
**isdn switch-type basic-ni**

## Syntax Description

| | |
|---|---|
| **basic-5ess** | Specifies Lucent/AT&T 5ESS signaling. |
| **basic-dms** | Specifies Nortel DMS-100 custom signaling.The **basic-dms** signaling type is not compatible with proprietary SL-1 DMS signaling. |
| **basic-net3** | Specifies Net3 Euro-ISDN signaling. |
| **basic-ni** | Specifies National ISDN-1 signaling. |

## Default Values

By default, the ISDN signaling is set to National ISDN-1.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

The **isdn switch-type** command specifies the type of ISDN signaling implemented on the BRI interface, not the manufacturer of the Central Office switch. It is quite possible to have a Lucent Central Office switch providing National ISDN signaling on the BRI interface.

## Usage Examples

The following example configures a BRI interface for a circuit with Lucent 5ESS (custom) signaling:

(config)**#interface bri 1/2**
(config-bri 1/2)**#isdn switch-type basic-5ess**

# loopback local

Use the **loopback local** command to enable a local loopback of the interface (towards the router). Use the **no** form of this command to disable the loopback. Variations of this command include:

**loopback local all**
**loopback local b1**
**loopback local b2**
**loopback local both**

## Syntax Description

| | |
|---|---|
| **all** | Loops the entire interface back towards the router (including the D-channel). With an active **loopback active all**, the established D-channel between the ISDN module and the central office switch drops. |
| **b1** | Loops the data on B1 back towards the router. A B1 loopback does not disrupt D-channel signaling. |
| **b2** | Loops the data on B2 back towards the router. A B2 loopback does not disrupt D-channel signaling. |
| **both** | Loops the data on B1 and B2 back towards the router, but does not disrupt D-channel signaling. |

## Default Values

No default necessary for this command.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example enables a B2 loopback of the BRI 1/2 interface and disables the loopback:

(config)**#interface bri 1/2**
(config-bri 1/2)**#loopback local b2**
(config-bri 1/2)**#no loopback local b2**

# loopback network

Use the **loopback network** command to enable a loopback of the interface (towards the network). Use the **no** form of this command to disable the loopback. Variations of this command include:

**loopback network b1**
**loopback network b2**
**loopback network both**

## Syntax Description

| | |
|---|---|
| **b1** | Loops the data on B1 back towards the network. A B1 loopback does not disrupt D-channel signaling. |
| **b2** | Loops the data on B2 back towards the network. A B2 loopback does not disrupt D-channel signaling. |
| **both** | Loops the data on B1 and B2 back towards the network, but does not disrupt D-channel signaling. |

## Default Values

No default necessary for this command.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example enables a B2 loopback of the BRI 1/2 interface and disables the loopback:

(config)**#interface bri 1/2**
(config-bri 1/2)**#loopback network b2**
(config-bri 1/2)**#no loopback network b2**

# maintenance

Use the **maintenance** command to force a reset of the interface (initiating the SABME/UA process) or to reset the D-channel (by sending a RESTART message). Variations of this command include:

**maintenance reset**
**maintenance restart-d**


*The **maintenance** command disrupts data flow on the ISDN interface. All active calls will drop when the reset or restart process begins.*

## Syntax Description

| | |
|---|---|
| **reset** | Forces a complete reset of the interface by initiating the SABME/UA process. |
| **restart-d** | Resets the D-channel by sending a Q.931 RESTART message to the central office switch. |

## Default Values

No default necessary for this command.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example resets the BRI 1/2 interface:

(config)**#interface bri 1/2**
(config-bri 1/2)**#maintenance reset**

# resource pool-member <*name*>

Use the **resource pool-member** command to assign the interface to a resource pool, making it a demand routing resource. Use the **no** form of this command to return to the default value. Variations of this command include:

**resource pool-member** <*name*>
**resource pool-member** <*name*> <*priority*>

## Syntax Description

| | |
|---|---|
| <*name*> | Specifies the name of the resource pool to which this interface is assigned. |
| <*priority*> | Optional. Specifies the priority value of using this interface versus other interfaces contained in the specified resource pool using a number 1 to 255. Lower numbers indicate higher priority. Interfaces with the same priority are selected in alphabetical order by interface name. |

## Default Values

By default, the interface is not assigned to any resource pool.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example configures a BRI interface as a member of resource pool **MyPool**:

(config)**#interface bri 1/2**
(config-bri 1/2)**#resource pool-member MyPool**

# DDS INTERFACE CONFIGURATION COMMAND SET

To activate the DDS Interface Configuration mode, enter the **interface dds** command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#interface dds 1/1
(config-dds 1/1)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:



All other commands for this command set are described in this section in alphabetical order.



# clock rate

Use the **clock rate** command to configure the data rate used as the operating speed for the interface. This rate should match the rate required by the DDS service provider. Use the **no** form of this command to return to the default value. Variations of this command include:

**clock rate auto**
**clock rate bps56k**
**clock rate bps64k**

## Syntax Description

| | |
|---|---|
| **auto** | Automatically detects the clock rate and sets to match. |
| **bps56k** | Sets the clock rate to 56 kbps. |
| **bps64k** | Sets the clock rate to 64 kbps. |

## Default Values

By default, the rate is set to **auto**.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

When operating at 64 kbps (clear channel operation), the DTE data sequences may mimic network loop maintenance functions and erroneously cause other network elements to activate loopbacks. Use the **data-coding scrambled** command to prevent such occurrences. Refer to *data-coding scrambled* on page 788 for related information.

## Usage Examples

The following example configures the clock rate for 56 kbps operation:

(config)**#interface dds 1/1**
(config-dds 1/1)**#clock rate bps56k**

# clock source

Use the **clock source** command to configure the source timing used for the interface. The clock specified using the **clock source** command is also the system master clock. Use the **no** form of this command to return to the default value. Variations of this command include:

**clock source internal**
**clock source line**

## Syntax Description

| | |
|---|---|
| **internal** | Configures the unit to provide clocking using the internal oscillator. |
| **line** | Configures the unit to recover clocking from the DDS circuit. |

## Default Values

By default, the clock source is set to **line**.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

When operating on a DDS network, the clock source should be **line**. On a point-to-point private network, one unit must be **line** and the other **internal**.

## Usage Examples

The following example configures the unit to recover clocking from the circuit:

(config)**#interface dds 1/1**
(config-dds 1/1)**#clock source line**

# data-coding scrambled

Use the **data-coding scrambled** command to enable the DDS OS scrambler to combine user data with pattern data to ensure user data does not mirror standard DDS loop codes. The scrambler may only be used on 64 kbps circuits without Frame Relay signaling (clear channel). Use the **no** form of this command to return to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, the scrambler is disabled.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

When operating at 64 kbps (clear channel operation), there is a possibility the DTE data sequences may mimic network loop maintenance functions and erroneously cause other network elements to activate loopbacks. Use the **data-coding scrambled** command to prevent such occurrences. Do not use this command if using Frame Relay or if using PPP to another device other than an AOS product also running scrambled.

## Usage Examples

The following example enables the DDS OS scrambler:

```
(config)#interface dds 1/1
(config-dds 1/1)#data-coding scrambled
```

# loopback

Use the **loopback** command to initiate a specified loopback on the interface. Use the **no** form of this command to deactivate the loop. Variations of this command include:

**loopback dte**
**loopback line**
**loopback remote**

## Syntax Description

| | |
|---|---|
| **dte** | Initiates a loop to connect the transmit and receive path through the unit. |
| **line** | Initiates a loop of the DDS circuit toward the network by connecting the transmit path to the receive path. |
| **remote** | Transmits a DDS loop code over the circuit to the remote unit. In response, the remote unit should initiate a line loopback. |

## Default Values

No default values necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example activates a line loopback on the DDS interface:

(config)**#interface dds 1/1**
(config-dds 1/1)**#loopback line**

# remote-loopback

Use the **remote-loopback** command to configure the interface to respond to loopbacks initiated by a remote unit (or the service provider). Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, all interfaces respond to remote loopbacks.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example enables remote loopbacks on the DDS interface:

(config)**#interface dds 1/1**
(config-dds 1/1)**#remote-loopback**

# snmp trap

Use the **snmp trap** command to enable all supported Simple Network Management Protocol (SNMP) traps on the interface. Use the **no** form of this command to return to disable this trap.

## Syntax Description

No subcommands.

## Default Values

By default, all interfaces (except virtual Frame Relay interfaces and sub-interfaces) have SNMP traps enabled.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 3.1 | Command was extended to the SHDSL interface. |
| Release 5.1 | Command was expanded to include Ethernet sub-interfaces and Gigabit Ethernet interfaces. |

## Usage Examples

The following example enables SNMP capability on the DDS interface:

(config)**#interface dds 1/1**
(config-dds 1/1)**#snmp trap**

# snmp trap link-status

Use the **snmp trap link-status** command to control the Simple Network Management Protocol (SNMP) variable ifLinkUpDownTrapEnable (RFC2863) to enable (or disable) the interface to send SNMP traps when there is an interface status change. Use the **no** form of this command to disable this trap.

## Syntax Description

No subcommands.

## Default Values

By default, the ifLinkUpDownTrapEnable OID is enabled for all supported interfaces except virtual Frame Relay interfaces.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 3.1 | Command was extended to the SHDSL interface. |
| Release 5.1 | Command was expanded to include Gigabit Ethernet, port channel, VLAN, E1, and G.703 interfaces. |

## Functional Notes

The **snmp trap link-status** command is used to control the RFC2863 ifLinkUpDownTrapEnable OID (OID number 1.3.6.1.2.1.31.1.1.1.14.0).

## Usage Examples

The following example disables the link-status trap on the DDS interface:

(config)**#interface dds 1/1**
(config-dds 1/1)**#no snmp trap link-stat**us

# DSX-1 INTERFACE CONFIGURATION COMMAND SET

To activate the DSX-1 Interface Configuration mode, enter the **interface t1** command (and specify the DSX-1 port) at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#interface t1 1/2
(config-t1 1/2)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:



All other commands for this command set are described in this section in alphabetical order.



# coding

Use the **coding** command to configure the line coding for a DSX-1 physical interface. This setting must match the line coding supplied on the circuit by the PBX. Use the **no** form of this command to return to the default setting. Variations of this command include:

**coding ami**
**coding b8zs**

## Syntax Description

| | |
|---|---|
| **ami** | Configures the line coding for alternate mark inversion (AMI). |
| **b8zs** | Configures the line coding for bipolar eight zero substitution (B8ZS). |

## Default Values

By default, all DSX-1 interfaces are configured with B8ZS line coding.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

The line coding configured in the unit must match the line coding of the DSX-1 circuit. A mismatch will result in line errors (e.g., BPVs).

## Usage Examples

The following example configures the DSX-1 interface for AMI line coding:

(config)**#interface t1 1/2**
(config-t1 1/2)**#coding ami**

# framing

Use the **framing** command to configure the framing format for the DSX-1 interface. This parameter should match the framing format set on the external device. Use the **no** form of this command to return to the default value. Variations of this command include:

**framing d4**
**framing esf**

## Syntax Description

| | |
|---|---|
| **d4** | Specifies D4 superframe (SF) format. |
| **esf** | Specifies extended superframe (ESF) format. |

## Default Values

By default, the framing format is set to **esf**.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

A frame is comprised of a single byte from each of the T1's timeslots; there are 24 timeslots on a single T1 circuit. Framing bits are used to separate the frames and indicate the order of information arriving at the receiving equipment. D4 and ESF are two methods of collecting and organizing frames over the circuit.

## Usage Examples

The following example configures the DSX-1 interface for D4 framing:

(config)**#interface t1 1/2**
(config-t1 1/2)**#framing d4**

# line-length *<value>*

Use the **line-length** command to set the line build out (in feet or dB) for the DSX-1 interface. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| *<value>* | Configures the line build out for the DSX-1 interface. Valid options include: -7.5 dB or 0 to 655 feet. Use the -7.5 dB option for maximum attenuation. |

## Default Values

By default, the line build out is set to 0 feet.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

The **line-length** value represents the physical distance between DSX equipment (measured in cable length). Based on this setting, the AOS device increases signal strength to compensate for the distance the signal must travel. Valid distance ranges are listed below:

- 0 to 133 feet
- 134 to 265 feet
- 266 to 399 feet
- 400 to 533 feet
- 534 to 655 feet

## Usage Examples

The following example configures the DSX-1 interface **line-length** for 300 feet:

(config)**#interface t1 1/2**
(config-t1 1/2)**#line-length 300**

# loopback network

Use the **loopback network** command to initiate a loopback on the interface toward the network. Use the **no** form of this command to deactivate the loopback. Variations of this command include:

**loopback network line**
**loopback network payload**

## Syntax Description

| | |
|---|---|
| **line** | Initiates a metallic loopback of the physical DSX-1 network interface. |
| **payload** | Initiates a loopback of the T1 framer (CSU portion) of the DSX-1 network interface. |

## Default Values

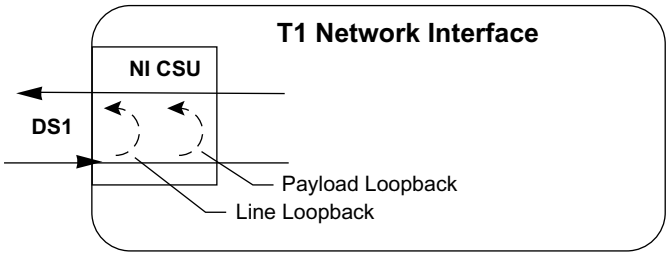
No default necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

The following diagram depicts the difference between a line and payload loopback.




## Usage Examples

The following example initiates a payload loopback of the DSX-1 interface:

(config)**#interface t1 1/2**
(config-t1 1/2)**#loopback network payload**

# loopback remote line inband

Use the **loopback remote line inband** command to send a loopback code to the remote unit to initiate a line loopback. Use the **no** form of this command to send a loopdown code to the remote unit to deactivate the loopback.

## Syntax Description

| | |
|---|---|
| **inband** | Uses the inband channel to initiate a full 1.544 Mbps physical (metallic) loopback of the signal received by the remote unit from the network. |

## Default Values

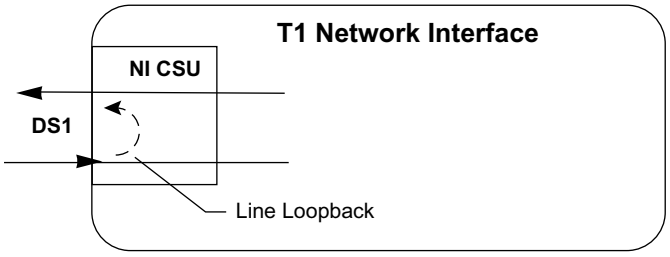
No defaults necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

A remote loopback can only be issued if a cross-connect does not exist on the interface and if the signaling mode is set to **none**. The following diagram depicts the difference between a line and payload loopback.




## Usage Examples

The following example initiates a remote line loopback using the inband channel:

(config)**#interface t1 1/2**
(config-t1 1/2)**#loopback remote line inband**

# remote-loopback

Use the **remote-loopback** command to configure the interface to respond to loopbacks initiated by a remote unit (or the service provider). Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, all interfaces respond to remote loopbacks.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example enables remote loopbacks on the DSX-1 interface:

(config)**#interface t1 1/2**
(config-t1 1/2)**#remote-loopback**

# signaling-mode

Use the **signaling-mode** command to configure the signaling type (robbed-bit for voice or clear channel for data) for the DS0s mapped to the DSX-1 port. Use the **no** form of this command to return to the default setting. Variations of this command include:

**signaling-mode message-oriented**
**signaling-mode none**
**signaling-mode robbed-bit**

## Syntax Description

| | |
|---|---|
| **message-oriented** | Specifies clear channel signaling on Channel 24 only. Use this signaling type with QSIG installations. |
| **none** | Specifies clear channel signaling on all 24 DS0s. Use this signaling type with data-only or PRI DSX-1 installations. |
| **robbed-bit** | Specifies robbed bit signaling on all DS0s. Use this signaling type for voice-only DSX-1 applications. |

## Default Values

By default, the signaling mode is set to **robbed-bit**.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example configures the DSX-1 port for PRI compatibility:

(config)**#interface t1 1/2**
(config-t1 1/2)**#signaling-mode none**

# snmp trap link-status

Use the **snmp trap link-status** command to control the Simple Network Management Protocol (SNMP) variable ifLinkUpDownTrapEnable (RFC2863) to enable (or disable) the interface to send SNMP traps when there is an interface status change. Use the **no** form of this command to disable this trap.

## Syntax Description

No subcommands.

## Default Values

By default, the ifLinkUpDownTrapEnable OID is enabled for all interfaces except virtual Frame Relay interfaces.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 3.1 | Command was extended to the SHDSL interface. |
| Release 5.1 | Command was expanded to include Gigabit-Ethernet, port-channel, VLAN, E1, and G.703 interfaces. |

## Functional Notes

The **snmp trap link-status** command is used to control the RFC2863 ifLinkUpDownTrapEnable OID (OID number 1.3.6.1.2.1.31.1.1.1.14.0).

## Usage Examples

The following example disables the link-status trap on the DSX-1 interface:

(config)**#interface t1 1/2**
(config-t1 1/2)**#no snmp trap link-status**

# test-pattern

Use the **test-pattern** command to activate the built-in pattern generator and begin sending the specified test pattern. This pattern generation can be used to verify a data path when used in conjunction with an active loopback. Use the **no** form of this command to cease pattern generation. Variations of this command include:

**test-pattern ones**
**test-pattern zeros**

## Syntax Description

| | |
|---|---|
| **ones** | Generates a test pattern of continous ones. |
| **zeros** | Generates a test pattern of continous zeros. |

## Default Values

No defaults necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example activates the pattern generator for a stream of continuous ones:

(config)**#interface t1 1/2**
(config-t1 1/2)**#test-pattern ones**

# E1 INTERFACE CONFIGURATION COMMAND SET

To activate the E1 Interface Configuration mode, enter the **interface e1** command (and specify the E1 port) at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#interface e1 1/1
(config-e1 1/1)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:



All other commands for this command set are described in this section in alphabetical order.



# clock source

Use the **clock source** command to configure the source timing used for the interface. Use the **no** form of this command to return to the default value. Variations of this command include:

**clock source internal**
**clock source line**
**clock source through**

## Syntax Description

| | |
|---|---|
| **internal** | Configures the unit to provide clocking using the internal oscillator. |
| **line** | Configures the unit to recover clocking from the E1 circuit. |
| **through** | Configures the unit to recover clocking from the circuit connected to the G.703 interface. |

## Default Values

By default, the unit is configured to recover clocking from the primary circuit.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

When operating on a circuit that is providing timing, setting the **clock source** to **line** can avoid errors such as Clock Slip Seconds (CSS).

## Usage Examples

The following example configures the unit to recover clocking from the primary circuit:

(config)**#interface e1 1/1**
(config-e1 1/1)**#clock source line**

# coding

Use the **coding** command to configure the line coding for the E1 physical interface. This setting must match the line coding supplied on the circuit by the service provider. Use the **no** form of this command to return to the default setting. Variations of this command include:

**coding ami**
**coding hdb3**

## Syntax Description

| | |
|---|---|
| **ami** | Configures the line coding for alternate mark inversion (AMI). |
| **hdb3** | Configures the line coding for high-density bipolar 3 (HDB3). |

## Default Values

By default, all E1 interfaces are configured with HDB3 line coding.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

The line coding configured in the unit must match the line coding of the E1 circuit. A mismatch will result in line errors (e.g., BPVs).

## Usage Examples

The following example configures the E1 interface for AMI line coding:

(config)**#interface e1 1/1**
(config-e1 1/1)**#coding ami**

# framing crc4

Use the **framing** command to configure the framing format for the E1 interface. This parameter should match the framing format provided by the service provider or external device. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| **crc4** | Enables CRC-4 bits to be transmitted in the outgoing data stream. Also, the received signal is checked for CRC-4 errors. |

## Default Values

By default, CRC-4 is disabled.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

The framing value must match the configuration of the E1 circuit. A mismatch will result in a loss of frame alarm.

## Usage Examples

The following example configur**es the E1 interface for CRC-4 framing:**

**(config)#**interface e1 1/1
(config-e1 1/1)**#framing crc4**

# loop-alarm-detect

The **loop-alarm-detect** command enables detection of a loop alarm on the E1 interface. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, this command is enabled.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Functional Notes

This command enables the detection of a loopback alarm. This alarm works in conjunction with the **sa4tx-bit** command setting. The loopback condition is detected by comparing the transmitted **sa4tx-bit** value to the received Sa4 bit value. If the bits match, a loopback is assumed. This detection method only works with a network in which the far end is transmitting the opposite value for Sa4.

## Usage Examples

The following example enables detection of a loop alarm on the E1 interface:

(config)**#config e1 1/1**
(config-e1 1/1)**#loop-alarm-detect**

# loopback network

Use the **loopback network** command to initiate a loopback on the interface toward the network. Use the **no** form of this command to deactivate the loopback. Variations of this command include:

**loopback network line**
**loopback network payload**

## Syntax Description

| | |
|---|---|
| **line** | Initiates a metallic loopback of the physical E1 network interface. |
| **payload** | Initiates a loopback of the E1 framer (CSU) portion of the E1 network interface. |

## Default Values

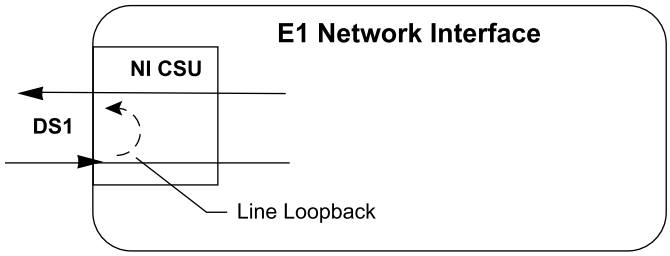
No default necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

The following diagram depicts a line loopback.




## Usage Examples

The following example initiates a line loopback of the E1 interface:

(config)**#interface e1 1/1**
(config-e1 1/1)**#loopback network line**

# loopback remote v54

The **loopback remote v54** command initiates an E1 remote loopback test (with a V.54 loopback pattern). Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

No default value is necessary for this command.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Functional Notes

This command causes a V.54 inband loop code to be sent in the payload towards the far end.

## Usage Examples

The following example sends a V.54 inband loop code to the far end:

(config)**#interface e1 1/1**
(config-e1 1/1)**#loopback remote v54**

# remote-alarm

The **remote-alarm** command selects the alarm signaling type to be sent when a loss of frame is detected on the E1 receive signal. Use the **no** form of this command to disable all transmitted alarms. Variations of this command include:

**remote-alarm rai**
**remote-alarm ais**

## Syntax Description

| | |
|---|---|
| **rai** | Specifies sending a remote alarm indication (RAI) in response to a loss of frame. Also prevents a received RAI from causing a change in interface operational status. |
| **ais** | Sends an alarm indication signal (AIS) as an unframed all-ones signal. |

## Default Values

The default for this command is RAI.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Functional Notes

An E1 will respond to a loss of frame on the receive signal by transmitting a remote alarm to the far end to indicate the error condition. TS0 of an E1 contains the Frame Alignment Signal (FAS) in the even-numbered frames. The odd-numbered frames are not used for frame alignment, and some of those bits are labeled as spare bits (Sa bits) in bit positions 4 through 8.

## Usage Examples

The following example enables transmission of AIS in response to a loss of frame:

(config)**#interface e1 1/1**
(config-e1 1/1)**#remote alarm ais**

# remote-loopback

Use the **remote-loopback** command to configure the interface to respond to loopbacks initiated by a remote unit (or the service provider). Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, all interfaces respond to remote loopbacks.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

This controls the acceptance of any remote loopback requests. When enabled, remote loopbacks are detected and cause a loopback to be applied. When disabled, remote loopbacks are ignored.

## Usage Examples

The following example enables remote loopbacks on the E1 interface:

(config)**#interface e1 1/1**
(config-e1 1/1)**#remote-loopback**

# sa4tx-bit <*value*>

The **sa4tx-bit** command selects the Tx value of Sa4 in this E1 interface. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies a 0 or a 1 for the transmit value of the SA4 bit on the E1. |

## Default Values

The default value for this command is 1.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Functional Notes

This command assigns a value to the Tx spare bit in position 4. The odd-numbered frames of TS0 are not used for frame alignment. Bits in position 4 through 8 are called spare bits. Values of 0 or 1 are accepted.

TS0 odd frame

| Bit position | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| Bit use | 0 | 1 | RAI = 1 | S | S | S | S | S |

## Usage Examples

The following example sets the Tx value of Sa4 to 0:

(config)**#interface e1 1/1**
(config-e1 1/1)**#sa4tx-bit 0**

# snmp trap line-status

Use the **snmp trap line-status** command to control the Simple Network Management Protocol (SNMP) variable dsx1LineStatusChangeTrapEnable (RFC2495) to enable (or disable) the interface to send SNMP traps when there is an interface status change. Use the **no** form of this command to disable this trap.

## Syntax Description

No subcommands.

## Default Values

By default, the dsx1LineStatusChangeTrapEnable OID is set to enabled for all interfaces except virtual Frame Relay Interfaces.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Functional Notes

The **snmp trap line-status** command is used to control the RFC2495 dsx1LineStatusChangeTrapEnable OID (OID number 1.3.6.1.2.1.10.18.6.1.17.0).

## Usage Examples

The following example disables the line-status trap on the T1 interface:

(config)**#interface e1 1/1**
(config-t1 1/1)**#no snmp trap line-status**

# snmp trap link-status

Use the **snmp trap link-status** command to control the Simple Network Management Protocol (SNMP) variable ifLinkUpDownTrapEnable (RFC2863) to enable (or disable) the interface to send SNMP traps when there is an interface status change. Use the **no** form of this command to disable this trap.

## Syntax Description

No subcommands.

## Default Values

By default, the ifLinkUpDownTrapEnable OID is enabled for all interfaces except virtual Frame Relay interfaces.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 3.1 | Command was extended to the SHDSL interface. |
| Release 5.1 | Command was expanded to include Gigabit Ethernet, port channel, VLAN, E1, and G.703 interfaces. |

## Functional Notes

The **snmp trap link-status** command is used to control the RFC 2863 ifLinkUpDownTrapEnable OID (OID number 1.3.6.1.2.1.31.1.1.1.14.0).

## Usage Examples

The following example disables the link-status trap on the E1 interface:

(config)**#interface e1 1/1**
(config-e1 1/1)**#no snmp trap link-status**

# snmp trap threshold-reached

Use the **snmp trap threshold-reached** command to control the Simple Network Management Protocol (SNMP) variable adGenAOSDs1ThresholdReached (adGenAOSDs1-Ext MIB) to enable the interface to send SNMP traps when a DS1 performance counter threshold is reached. Use the **no** form of this command to disable this trap.

## Syntax Description

No subcommands.

## Default Values

By default, the adGenAOSDs1ThresholdReached OID is enabled for all interfaces.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example disables SNMP threshold reached trap on the E1 interface:

(config)**#interface e1 1/1**
(config-e1 1/1)**#no snmp trap threshold-reached**

# tdm-group *<number>*

Use the **tdm-group** command to create a group of contiguous channels on this interface to be used during the **cross-connect** process. Use the **no** form of this command to remove configured TDM groups. Refer to *cross-connect* on page 29 for related information. Variations of this command include:

**tdm-group** *<number>* **timeslots** *<value>*
**tdm-group** *<number>* **timeslots** *<value>* **speed [56 | 64]**


*Changing **tdm-group** settings could result in service interruption.*

## Syntax Description

| | |
|---|---|
| *<number>* | Identifies the created TDM group. Valid range is 1 to 255. |
| **timeslots** *<value>* | Specifies the channels to be used in the TDM group. Valid range is 1 to 31. The timeslot value can be entered as a single number representing one of the 31 E1 channel timeslots or as a contiguous group of channels. (For example, **1-10** specifies the first 10 channels of the E1.) |
| **speed [56 \| 64]** | Optional. Specifies the individual channel rate on the E1 interface to be 56 or 64 kbps. The default speed is 64 kbps. 56 kbps operation is not available on all E1 interfaces. Refer to the *Quick Start Guide* provided with your E1 module to determine whether 56 kbps is valid. |

## Default Values

By default, there are no configured TDM groups.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following example creates a TDM group (labeled **5**) of 10 channels at 64 kbps each:

(config)**#interface e1 1/1**
(config-e1 1/1)**#tdm-group 5 timeslots 1-10 speed 64**

# test-pattern

Use the **test-pattern** command to activate the built-in pattern generator and begin sending the specified test pattern. This pattern generation can be used to verify a data path when used in conjunction with an active loopback. Use the **no** form of this command to cease pattern generation. Variations of this command include:

**test-pattern clear**
**test-pattern errors**
**test-pattern insert**
**test-pattern ones**
**test-pattern p215**
**test-pattern p220**
**test-pattern p511**
**test-pattern qrss**
**test-pattern zeros**

## Syntax Description

| | |
|---|---|
| **clear** | Clears the test pattern error count. |
| **errors** | Displays the test pattern error count. |
| **insert** | Inserts an error into the currently active test pattern. |
| **ones** | Generates test pattern of continous ones. |
| **p215** | Generates a pseudorandom test pattern sequence based on a 15-bit shift register. |
| **p220** | Generates a pseudorandom test pattern sequence based on a 20-bit shift register. |
| **p511** | Generates a test pattern of repeating ones and zeros. |
| **qrss** | Generates a test pattern of random ones and zeros. |
| **zeros** | Generates test pattern of continous zeros. |

## Default Values

No defaults necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following example activates the pattern generator for a stream of continuous ones:

(config)**#interface e1 1/1**
(config-e1 1/1)**#test-pattern ones**

# ts16

Use the **ts16** command to enable timeslot 16 multiframe to be checked on the receive signal. Use the **no** form of this command to disable timeslot 16.

## Syntax Description

No subcommands.

## Default Values

No defaults necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

If timeslot 16 is used on the incoming E1, do not map timeslot 16 using the **tdm-group** command. By default, all timeslots not physically mapped using the **tdm-group** command are passed through to the G.703 interface. Leaving timeslot 16 unmapped makes it available for multiframe signaling by the connected E1 device.

## Usage Examples

The following example enables timeslot 16 multiframing:

(config)**#interface e1 1/1**
(config-e1 1/1)**#ts16**

# ETHERNET INTERFACE CONFIGURATION COMMAND SET

There are four types of Ethernet interfaces associated with the AOS:

- Basic Ethernet interfaces (e.g., eth 0/1)
- Gigabit Ethernet interfaces (e.g., giga-eth 0/3)
- Ethernet sub-interfaces associated with a VLAN (e.g., eth 0/1.1)
- Ethernet interface range (e.g., eth 0/1, 0/8)

To activate the basic Ethernet Interface Configuration mode, enter the **interface ethernet** command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#interface ethernet 0/1
(config-eth 0/1)#
```

To activate the Gigabit Ethernet Interface Configuration mode, enter the **interface gigabit-ethernet** command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#interface gigabit-ethernet 0/3
(config-giga-eth 0/3)#
```

To activate the Ethernet Sub-Interface Configuration mode, enter the **interface ethernet** command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#interface ethernet 0/1.1
(config-eth 0/1.1)#
```

To activate the Ethernet Configuration mode for a range of Ethernet interfaces, enter the **interface range** command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#interface range ethernet 0/1, 0/8
(config-eth 0/1, 0/8)#
```

*Not all Ethernet commands apply to all Ethernet types. Use the ? command to display a list of valid commands. For example:*

*>**enable***
*Password:**xxxxx***
*#config term*
*(config)#**interface ethernet 0/1***
*(config-eth 0/1)#**?***

| | |
|---|---|
| *access-policy* | *- Assign access control policy for this interface* |
| *alias* | *- A text name assigned by an SNMP NMS* |
| *arp* | *- Set ARP commands* |
| *bandwidth* | *- Set bandwidth informational parameter* |
| *bridge-group* | *- Assign the current interface to a bridge group* |
| *etc....* | |

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:

*alias <"text"> on page 28*
*cross-connect on page 29*
*description <text> on page 32*
*do on page 33*
*end on page 34*
*exit on page 35*
*shutdown on page 36*

All other commands for this command set are described in this section in alphabetical order.

*access-policy <name> on page 822*
*arp arpa on page 823*
*bandwidth <value> on page 824*
*bridge-group <number> on page 825*
*bridge-group <number> vlan-transparent on page 826*
*channel-group <number> mode on on page 827*
*crypto map <name> on page 828*
*dynamic-dns on page 830*
*encapsulation 802.1q on page 832*
*full-duplex on page 833*
*half-duplex on page 834*
*ip commands begin on page 835*
*lldp receive on page 870*

# access-policy *<name>*

Use the **access-policy** command to assign a specified access policy to an interface. Access policies are applied to traffic entering an interface. Use the **no** form of this command to remove an access policy association. For more information on using access policies, refer to *ip policy-class <name>* on page 549.


*Configured access policies will only be active if the **ip firewall** command has been entered at the Global Configuration mode prompt to enable the AOS security features. All configuration parameters are valid, but no security data processing will be attempted unless the security features are enabled.*

## Syntax Description

| | |
|---|---|
| *<name>* | Identifies the configured access policy by alphanumeric descriptor (all access policy descriptors are case-sensitive). |

## Default Values

By default, there are no configured access policies associated with an interface.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include Ethernet sub-interfaces. |
| Release 6.1 | Command was expanded to include VLAN interfaces. |

## Functional Notes

To assign an access policy to an interface, enter the interface configuration mode for the desired interface and enter **access-policy** *<policy name>*.

## Usage Examples

The following example associates the access policy **Private** (to allow inbound traffic to the Web server) to the Ethernet 0/1 interface:

Enable the AOS security features:
(config)**#ip firewall**

Associate the access policy with the Ethernet 0/1 interface:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#access-policy Private**

# arp arpa

Use the **arp arpa** command to set ARPA as the standard address resolution protocol (ARP) on this interface. Use the **no** form of this command to return to the default setting.

## Syntax Description

No subcommands.

## Default Values

The default for this command is **arpa**.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include Ethernet sub-interfaces and Gigabit Ethernet interfaces. |

## Usage Examples

The following example enables standard ARP for the Ethernet interface:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#arp arpa**

# bandwidth *<value>*

Use the **bandwidth** command to provide the bandwidth value of an interface to the higher-level protocols. This value is used in cost calculations. Use the **no** form of this command to restore the default value.

## Syntax Description

| | |
|---|---|
| *<value>* | Specifies bandwidth in kbps. Range is 1 to 4,294,967,295 kbps. |

## Default Values

To view default value, use the **show interfaces** command.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include Ethernet sub-interfaces. |

## Functional Notes

The **bandwidth** command is an informational value that is communicated to the higher-level protocols to be used in cost calculations. This is a routing parameter only and does not affect the physical interface.

## Usage Examples

The following example sets bandwidth of the Ethernet 0/1 interface to 10 Mbps:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#bandwidth 10000**

# bridge-group *<number>*

Use the **bridge-group** command to assign an interface to the specified bridge group. Use the **no** form of this command to remove the interface from the bridge group.

## Syntax Description

| | |
|---|---|
| *<number>* | Specifies the bridge group (by number) to which to assign this interface. Range is 1 to 255. |

## Default Values

By default, there are no configured bridge groups.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

A bridged network can provide excellent traffic management to reduce collisions and limit the amount of bandwidth wasted with unnecessary transmissions when routing is not necessary. Any two interfaces can be bridged (e.g., Ethernet to T1 bridge, Ethernet to Frame Relay sub-interface).

## Usage Examples

The following example assigns the Ethernet interface to bridge-group 17:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#bridge-group 17**

# bridge-group <*number*> vlan-transparent

Use the **bridge-group vlan-transparent** command to prevent an interface from removing the VLAN tag. Use the **no** form of this command to allow the interface to remove the VLAN tag from the packet.

*The **bridge-group vlan-transparent** command is not a global command. The command must be applied on all interfaces of the bridge group.*

## Syntax Description

| | |
|---|---|
| <*number*> | Specifies the bridge group number. Valid range is 1 to 255. |

## Default Values

By default, VLAN tags are removed from the data.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example removes the VLAN tags from the packets on the Ethernet interface 0/1:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#bridge-group 1 vlan-transparent**

# channel-group <*number*> mode on

Use the **channel-group mode on** command to statically add the interface to a channel-group. To remove an interface from a channel-group, use the **no** version of this command.

## Syntax Description

| | |
|---|---|
| <*number*> | Specifies the channel-group number. Range is 1 to 6. |

## Default Values

By default, the interface is not part of a channel group.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

There can be up to six channel groups with 2-8 interfaces per group. Dynamic protocols are not yet supported (only static). A physical interface can be a member of only one channel group.

## Usage Examples

The following example adds the Ethernet 0/1 interface to channel group 1:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#channel-group 1 mode on**
(config-eth 0/1)#

# crypto map *<name>*

Use the **crypto map** command to associate crypto maps with the interface. Use the **no** form of this command to remove a crypto map from an interface.

> *When you apply a map to an interface, you are applying all crypto maps with the given map name. This allows you to apply multiple crypto maps if you have created maps which share the same name but have different map index numbers.*

> *For VPN configuration example scripts, refer to the technical support note* ***Configuring VPN*** *located on the ADTRAN OS System Documentation CD provided with your unit.*

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies the crypto map name that you wish to assign to the interface. |

## Default Values

By default, no crypto maps are assigned to an interface.

## Command History

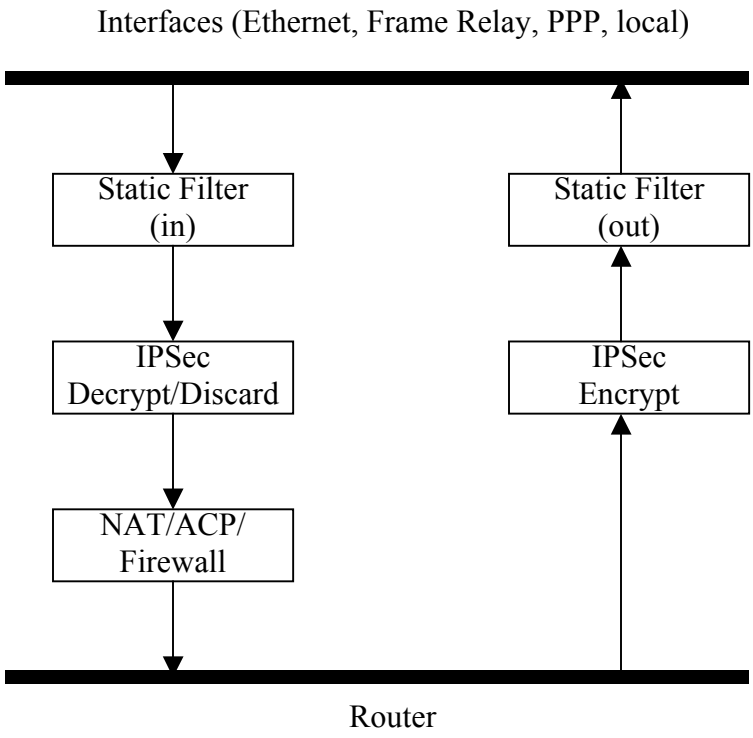
| | |
|---|---|
| Release 4.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include Ethernet sub-interfaces. |

## Functional Notes

When configuring a system to use both the stateful inspection firewall and IKE negotiation for VPN, keep the following notes in mind.

When defining the policy class and associated access-control lists (ACLs) that describe the behavior of the firewall, do not forget to include the traffic coming into the system over a VPN tunnel terminated by the system. The firewall should be set up with respect to the unencrypted traffic that is destined to be sent or received over the VPN tunnel. The following diagram represents typical AOS data-flow logic.




As shown in the diagram above, data coming into the product is first processed by the static filter associated with the interface on which the data is received. This access group is a true static filter and is available for use regardless of whether the firewall is enabled or disabled. Next (if the data is encrypted) it is sent to the IPSec engine for decryption. The decrypted data is then processed by the stateful inspection firewall. Therefore, given a terminating VPN tunnel, only unencrypted data is processed by the firewall.

The ACLs for a crypto map on an interface work in reverse logic to the ACLs for a policy class on an interface. When specifying the ACLs for a crypto map, the source information is the private local side, unencrypted source of the data. The destination information will be the far end, unencrypted destination of the data. However, ACLs for a policy class work in reverse. The source information for the ACL in a policy class is the far end. The destination information is the local side.

## Usage Examples

The following example applies all crypto maps with the name **MyMap** to the Ethernet interface:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#crypto map MyMap**

# dynamic-dns

Use the **dynamic-dns** command to configure Dynamic DNS service provided by Dynamic Network Services, Inc. (www.dyndns.org). Use the **no** versions of these commands to disable these features. Variations of this command include:

**dynamic-dns custom** <*hostname*> <*minutes*>
**dynamic-dns dyndns** <*hostname*> <*username*> <*password*>
**dynamic-dns dyndns-custom** <*hostname*> <*username*> <*password*>
**dynamic-dns dyndns-static** <*hostname*> <*username*> <*password*>

## Syntax Description

| | |
|---|---|
| <*hostname*> | Specifies the host name for the server that updates the Dynamic Domain Name Server (DNS). |
| <*minutes*> | Specifies the intervals in minutes to update the server with information (updates also occur when the interface's IP address changes regardless of the update intervals).<br>Refer to *Functional Notes* below for additional argument descriptions. |

## Default Values

No default is necessary for this command.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |
| Release 12.1 | Command was expanded. |

## Functional Notes

**custom** - Constanttime.com's Custom Dynamic DNS$^{SM}$ service allows you complete access and management control over your domain name regardless of where you purchased/registered it. This allows to manage IP address mappings (A records), domain aliases (CNAME records) and mail servers (MX records).

**dyndns** - The Dynamic DNS$^{SM}$ offered by Dynamic Network Services Inc., (DynDNS.org) allows you to alias a dynamic IP address to a static host name in various domains. This allows your unit to be more easily accessed from various locations on the Internet. This service is provided for up to five host names.

**dyndns-custom -** DynDNS.org's Custom DNS$^{SM}$ service provides a full DNS solution, giving you complete control over an entire domain name. A web-based interface provides two levels of control over your domain, catering to average or advanced users. Five globally redundant DNS servers ensure that your domain will always resolve.

A choice of two interfaces is available. The basic interface is designed for most users. It comes preconfigured for the most common configuration and allows for easy creation of most common record types. The advanced interface is designed for system administrators with a solid DNS background, and provides layout and functionality similar to a BIND zone file allowing for the creation of nearly any record type.

Custom DNS$^{SM}$ can be used with both static and dynamic IPs and has the same automatic update capability through Custom DNS-aware clients as Dynamic DNS.

**dyndns-static -** The Static DNS service is similar to DynDNS.org's Dynamic DNS$^{SM}$ service in that it allows a host name such as yourname.dyndns.org to point to your IP address. Unlike a Dynamic DNS host, a Static DNS host does not expire after 35 days without updates, but updates take longer to propagate though the DNS system. This service is provided for up to five host names.

If your IP address does not change often or at all but you still want an easy name to remember it by (without having to purchase your own domain name), Static DNS service is ideal for you.

If you would like to use your own domain name (such as yourname.com), you need Custom DNS service which also provides full dynamic and static IP address support.

**Usage Examples**

The following example sets the dynamic-dns to **dyndns-custom** with host name **host**, user's name **user**, and password **pass**:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#dynamic-dns dyndns-custom host user pass**

# encapsulation 802.1q

Use the **encapsulation 802.1q** command to put the interface into 802.1q (VLAN) mode.

## Syntax Description

No subcommands.

## Default Values

No default value is necessary for this command.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Usage Examples

The following example puts Ethernet interface 0/1 in 802.1q mode and configures a sub-interface for VLAN usage:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#encapsulation 802.1q**
(config-eth 0/1)**#interface ethernet 0/1.1**
(config-eth 0/1.1)**#vlan-id 3**

# full-duplex

Use the **full-duplex** command to configure the Ethernet interface for full-duplex operation. This allows the interface to send and receive simultaneously. Use the **no** form of this command to return to the default **half-duplex** operation.

## Syntax Description

No subcommands.

## Default Values

By default, all Ethernet interfaces are configured for half-duplex operation.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

Full-duplex Ethernet is a variety of Ethernet technology currently being standardized by the IEEE. Because there is no official standard, vendors are free to implement their independent versions of full-duplex operation. Therefore, it is not safe to assume that one vendor's equipment will work with another.

Devices at each end of a full-duplex link have the ability to send and receive data simultaneously over the link. Theoretically, this simultaneous action can provide twice the bandwidth of normal (half-duplex) Ethernet. To deploy full-duplex Ethernet, each end of the link must only connect to a single device (a workstation or a switched hub port). With only two devices on a full-duplex link, there is no need to use the medium access control mechanism (to share the signal channel with multiple stations) and listen for other transmissions or collisions before sending data.


*Some Ethernet equipment (though rare) is unable to negotiate duplex if speed is manually determined. To avoid incompatibilities, manually set the duplex if the speed is manually set.. Refer to speed* *on page 893* *for more information.*

The 10BaseT, 100BaseTX, and 100BaseFX signalling systems support full-duplex operation (because they have transmit and receive signal paths that can be simultaneously active).

## Usage Examples

The following example configures the Ethernet interface for **full-duplex** operation:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#full-duplex**

# half-duplex

Use the **half-duplex** command to configure the Ethernet interface for half-duplex operation. This setting allows the Ethernet interface to either send or receive at any given moment, but not simultaneously. Use the **no** form of this command to disable half-duplex operation.

## Syntax Description

No subcommands.

## Default Values

By default, all Ethernet interfaces are configured for half-duplex operation.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

Half-duplex Ethernet is the traditional form of Ethernet that employs the Carrier Sense Multiple Access/Collision Detect (CSMA/CD) protocol to allow two or more hosts to share a common transmission medium while providing mechanisms to avoid collisions. A host on a half-duplex link must “listen” on the link and only transmit when there is an idle period. Packets transmitted on the link are broadcast (so it will be “heard” by all hosts on the network). In the event of a collision (two hosts transmitting at once), a message is sent to inform all hosts of the collision and a backoff algorithm is implemented. The backoff algorithm requires the station to remain silent for a random period of time before attempting another transmission. This sequence is repeated until a successful data transmission occurs.


*Some Ethernet equipment (though rare) is unable to negotiate duplex if speed is manually determined. To avoid incompatibilities, manually set the duplex if the speed is manually set. Refer to speed* *on page 893* *for more information.*

## Usage Examples

The following example configures the Ethernet interface for **half-duplex** operation:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#half-duplex**

# ip access-group *<name>*

Use the **ip access-group** command to create an access control list to be used for packets transmitted on or received from the specified interface. Use the **no** form of this command to disable this type of control. Variations of this command include:

**ip access-group** <*name*> **in**
**ip access-group** <*name*> **out**

## Syntax Description

| | |
|---|---|
| *<name>* | Assigns IP access control list name. |
| **in** | Enables access control on packets received on the specified interface. |
| **out** | Enables access control on packets transmitted on the specified interface. |

## Default Values

By default, these commands are disabled.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include Ethernet sub-interfaces. |

## Functional Notes

When this command is enabled, the IP destination address of each packet must be validated before being passed through. If the packet is not acceptable per these settings, it is dropped.

## Usage Examples

The following example sets up the router to only allow Telnet traffic (as defined in the user-configured **TelnetOnly** IP access control list) into the Ethernet interface:

(config)**#ip access-list extended TelnetOnly**
(config-ext-nacl)**#permit tcp any any eq telnet**
(config-ext-nacl)**#interface ethernet 0/1**
(config-eth 0/1)**#ip access-group TelnetOnly in**

# ip address dhcp

Use the **ip address dhcp** command to use Dynamic Host Configuration Protocol (DHCP) to obtain an address on the interface. Use the **no** form of this command to remove a configured IP address (using DHCP) and disable DHCP operation on the interface. Variables that may be used with this command to further define the DHCP configuration include:

**ip address dhcp client-id [**<*interface*> **|** <*identifier*>**] [hostname** <*"string"*>**] [track** <*name*>**]**
**[**<*administrative distance*>**]**
**ip address dhcp hostname** <*"string"*> **[no-default-route | no-domain-name | no-nameservers] [track** <*name*>**] [**<*administrative distance*>**]**
**ip address dhcp [no-default-route | no-domain-name | no-nameservers] [track** <*name*>**]**
**[**<*administrative distance*>**]**
**ip address dhcp track** <*name*> **[**<*administrative distance*>**]**

## Syntax Description

| | |
|---|---|
| <*administrative distance*> | Optional. Specifies the administrative distance to use when adding the DHCP gateway into the route table. It is used to determine the best route when multiple routes to the same destination exist. The smaller the administrative distance the more reliable the route. Range is 1 to 255. |
| **client-id** | Optional. Specifies the client identifier used when obtaining an IP address from a DHCP server. |
| <*interface*> | Specifies an interface, thus defining the client identifier as the hexadecimal MAC address of the specified interface (including a hexadecimal number added to the front of the MAC address to identify the media type).<br>For example, specifying the **client-id ethernet 0/1** (where the Ethernet interface has a MAC address of d217.0491.1150) defines the client identifier as **01:d2:17:04:91:11:50** (where 01 defines the media type as Ethernet). Refer to *hardware-address <mac address>* on page 1973 for a detailed listing of media types. |
| <*identifier*> | Specifies a custom client-identifier using a text string (that is converted to a hexadecimal equivalent) or 7 to 28 hexadecimal numbers (with colon delimiters).<br>For example, a custom client identifier of **0f:ff:ff:ff:ff:51:04:99:a1** may be entered using the <*identifier*> option. |
| **hostname** <*"string"*> | Optional. Specifies a text string (to override the global router name) to use as the name in the DHCP option 12 field. The string is enclosed in quotation marks and can consist of up to 35 characters. |
| **no-default-route** | Optional. Specifies that no default route is obtained via DHCP. |
| **no-domain-name** | Optional. Specifies that no domain name is obtained via DHCP. |
| **no-nameservers** | Optional. Specifies that no DNS servers are obtained via DHCP. |
| **track** <*name*> | Optional. Attaches a network monitoring track to the DHCP client. The DHCP gateway route for this client will only reside in the route table while the track is in the pass state. For more information on configuring track objects, refer to *track <name>* on page 691. |

## Default Values

| | |
|---|---|
| *<administrative distance>* | By default, the administrative distance value is 1. |
| **client-id** | Optional. By default, the client identifier is populated using the following formula:<br>TYPE: INTERFACE SPECIFIC INFO : MAC ADDRESS<br>Where TYPE specifies the media type in the form of one hexadecimal byte (refer to *hardware-address <mac address>* on page 1973 for a detailed listing of media types), and the MAC ADDRESS is the Media Access Control (MAC) address assigned to the first Ethernet interface in the unit in the form of six hexadecimal bytes. (For units with a single Ethernet interface, the MAC ADDRESS assigned to Ethernet 0/1 is used in this field).<br>INTERFACE SPECIFIC INFO is only used for Frame Relay interfaces and can be determined using the following:<br>FR_PORT#: Q.922 ADDRESS<br>Where the FR_PORT# specifies the label assigned to the virtual Frame Relay interface using four hexadecimal bytes. For example, a virtual Frame Relay interface labeled 1 would have a FR_PORT# of 00:00:00:01.<br>The Q.922 ADDRESS field is populated using the following: |

| 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
|---|---|---|---|---|---|---|---|
| DLCI (high order) | | | | | | C/R | EA |
| DLCI (lower) | | | | FECN | BECN | DE | EA |

Where the FECN, BECN, C/R, DE, and high order extended address (EA) bits are assumed to be 0 and the lower order EA bit is set to 1.

The following list provides a few example DLCIs and associated Q.922 address:

DLCI (decimal) / Q.922 address (hex)

16 / 0x0401
50 / 0x0C21
60 / 0x0CC1
70 / 0x1061
80 / 0x1401

| | |
|---|---|
| **hostname** <*"string"*> | By default, the host name is the name configured using the Global Configuration **hostname** command. |

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |
| Release 8.1 | Command was expanded to include ATM sub-interface. |
| Release 13.1 | Command was expanded to include track and administrative distance. |

## Functional Notes

DHCP allows interfaces to acquire a dynamically assigned IP address from a configured DHCP server on the network. Many Internet Service Providers (ISPs) require the use of DHCP when connecting to their services. Using DHCP reduces the number of dedicated IP addresses the ISP must obtain. Consult your ISP to determine the proper values for the **client-id** and **hostname** fields.

## Usage Examples

The following example enables DHCP operation on the Ethernet interface 0/1:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#ip address dhcp**

The following example enables DHCP operation on the Ethernet interface 0/1 utilizing hostname **adtran** and does not allow obtaining a default route, domain name, or nameservers. It also sets the administrative distance as **5**:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#ip address dhcp hostname “adtran” no-default-route no-domain-name no-nameservers 5**

# ip address *<ip address> <subnet mask>*

Use the **ip address** command to define an IP address on the specified interface (only one primary address is allowed). Use the optional **secondary** keyword to define a secondary IP address. Use the **no** form of this command to remove a configured IP address. Variations of this command include:

**ip address** *<ip address> <subnet mask>*
**ip address** *<ip address> <subnet mask>* **secondary**

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies a valid IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| *<subnet mask>* | Specifies the subnet mask that corresponds to a range of IP addresses (network) or a specific host. Subnet masks can be expressed in dotted decimal notation (for example, **255.255.255.0**) or as a prefix length (for example, **/24**). |
| **secondary** | Optional. Configures a secondary IP address for the specified interface. |

## Default Values

By default, there are no assigned IP addresses.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include Ethernet sub-interfaces. |

## Functional Notes

Use secondary IP addresses to allow dual subnets on a single interface (when you need more IP addresses than the primary subnet can provide). When using secondary IP addresses, avoid routing loops by verifying that all devices on the network segment are configured with secondary IP addresses on the secondary subnet.

## Usage Examples

The following example configures a secondary IP address of **192.22.72.101/30**:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#ip address 192.22.72.101 255.255.255.252 secondary**

# ip dhcp release

Use the **ip dhcp release** command to transmit a message to the DHCP server requesting termination of the IP address lease on that interface.


*If you are currently connected to the unit using a Telnet session through the Ethernet interface, using the **ip dhcp release** command will terminate your Telnet session and render your Telnet capability inoperable until a new IP address is assigned by the DHCP server.*

## Syntax Description

No subcommands.

## Default Values

No defaults necessary for this command.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |

## Functional Notes

Dynamic Host Configuration Protocol (DHCP) allows interfaces to acquire a dynamically-assigned IP address from a configured DHCP server on the network. Many Internet Service Providers (ISPs) require the use of DHCP when connecting to their services. Using DHCP reduces the number of dedicated IP addresses the ISP must obtain.

## Usage Examples

The following example releases the IP address assigned (by DHCP) on the Ethernet interface 0/1:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#ip dhcp release**

# ip dhcp renew

Use the **ip dhcp renew** command to transmit a message to the DHCP server requesting renewal of the IP address lease on that interface.

## Default Values

No defaults necessary for this command.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |

## Functional Notes

Dynamic Host Configuration Protocol (DHCP) allows interfaces to acquire a dynamically assigned IP address from a configured DHCP server on the network. Many Internet Service Providers (ISPs) require the use of DHCP when connecting to their services. Using DHCP reduces the number of dedicated IP addresses the ISP must obtain.

## Usage Examples

The following example renews the IP address assigned (by DHCP) on the Ethernet interface 0/1:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#ip dhcp renew**

# ip ffe

Use the **ip ffe** command to enable the FastFlow Engine (FFE) on this interface with the default number of entries. Use the **no** form of these commands to disable the FFE. Variations of this command include:

**ip ffe max-entries** <*entries*>

*Issuing this command will cause all FFE entries on this interface to be cleared.*

## Syntax Description

| | |
|---|---|
| **max-entries** <*entries*> | Specifies the maximum number of entries stored in the flow table. Valid range is from 1 to 8192. |

## Default Values

By default, the FFE is disabled. The default number of **max-entries** is 4096.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Functional Notes

The FFE can be used to help reduce routing overhead and thus reduce overall routing times. Routing times are reduced by the creation of a flow table on the ingress interface. The maximum number of entries that can be stored in the flow table at any one time may be specified by using the **max-entries** subcommand. When features such as firewall, VPN, and static filters are enabled, routing time reduction may not be realized.

## Usage Examples

The following example enables the FFE and sets the maximum number of entries in the flow table to 50:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#ip ffe max-entries 50**

## Technology Review

The FastFlow system goal is to increase IP packet throughput by moving as much of the packet processing into the engine as possible. Packets are classified into flows based upon the IP protocol (TCP, UDP, ICMP, etc.), the source and destination IP addresses, IP TOS, and the protocol-specific information such as the source and destination port numbers. Flows are defined as the unidirectional representation of a conversation between two IP hosts. Each ingress interface keeps its own flow table, a collection of flow entries.

The first packet in a flow that is forwarded through the unit will build a flow entry. When a flow entry is looked up but no entry is found, a FastFlowBuilder object is allocated and attached to the packet. As the packet passes through the various processing layers, each subsystem will add processing to the FastFlowBuilder. When packet is about to be forwarded out of the egress interface, the FastFlowBuilder will be finalized. That is, the flow entry being built will be checked for completeness and committed to the flow table on the ingress interface. Subsequent flow matches can then bypass the normal processing layers.

# ip helper-address *<ip address>*

Use the **ip helper-address** command to configure the AOS to forward User Datagram Protocol (UDP) broadcast packets received on the interface. Use the **no** form of this command to disable forwarding packets.

*The **ip helper-address** command must be used in conjunction with the **ip forward-protocol** command to configure the AOS to forward UDP broadcast packets.*

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies the destination IP address for the forwarded UDP packets. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

## Default Values

By default, broadcast UDP packets are not forwarded.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include Ethernet sub-interfaces. |

## Functional Notes

When used in conjunction with the **ip forward-protocol** command, the **ip helper-address** feature allows you to customize which broadcast packets are forwarded.

To implement the helper address feature, assign helper address(es) (specifying the device that needs to receive the broadcast traffic) to the interface closest to the host that transmits the broadcast packets. When broadcast packets (of the specified type forwarded using the **ip forward-protocol** command) are received on the interface, they will be forwarded to the device that needs the information.

Only packets meeting the following criteria are considered eligible by the **ip helper-address** feature:

1. The packet IP protocol is User Datagram Protocol (UDP).

2. Any UDP port specified using the **ip forward-protocol** command.

3. The media access control (MAC) address of the frame is an all-ones broadcast address (ffff.ffff.ffff).

4. The destination IP address is broadcast defined by all ones (255.255.255.255) or a subnet broadcast (for example, 192.33.4.251 for the 192.33.4.248/30 subnet).

## Usage Examples

The following example forwards all DNS broadcast traffic to the DNS server with IP address **192.33.5.99**:

(config)**#ip forward-protocol udp domain**
**(config)#interface ethernet 0/1**
(config-eth 0/1)**#ip helper-address 192.33.5.99**

# ip igmp

Use the **ip igmp** command to configure multicasting-related functions for the interface. Variations of this command include:

**ip igmp immediate-leave**
**ip igmp last-member-query-interval** <*milliseconds*>
**ip igmp querier-timeout** <*seconds*>
**ip igmp query-interval** <*seconds*>
**ip igmp query-max-response-time** <*seconds*>
**ip igmp static-group** <*address*>
**ip igmp version [1 | 2]**

## Syntax Description

| | |
|---|---|
| **immediate-leave** | Specifies that if only one host (or IGMP snooping switch) is connected to the interface, when a leave is received, multicast of that group is immediately terminated as opposed to sending a group query and timing out the group if no device responds. Works in conjunction with **ip igmp last-member-query-interval**. Applies to all groups when configured. Use the **no** form of this command to disable the **immediate-leave** feature. |
| **last-member-query-interval** <*milliseconds*> | Controls the timeout (in milliseconds) used to detect whether any group receivers remain on an interface after a receiver leaves a group. If a receiver sends a leave-group message (IGMP Version 2), the router sends a group-specific query on that interface. After twice the time specified by this command plus as much as one second longer, if no receiver responds, the router removes that interface from the group and stops sending that group's multicast packets to the interface. Range is 100 to 65,535 ms. Use the **no** form of this command to return to the default setting. |
| **querier-timeout** <*seconds*> | Specifies the interval (in seconds) that the router waits after the current querier's last query before it takes over as querier (IGMP V2). Range is 60 to 300 seconds. Use the **no** form of this command to return to the default setting. |
| **query-interval** <*seconds*> | Specifies the interval (in seconds) at which IGMP queries are sent on an interface. Host query messages are addressed to the all-hosts multicast group with an IP TTL of 1. The router uses queries to detect whether multicast group members are on the interface and to select an IGMP designated router for the attached segment (if more than one multicast router exists). Only the designated router for the segment sends queries. For IGMP V2, the designated router is the router with the lowest IP address on the segment. Range is 0 to 65,535 seconds. Use the **no** form of this command to return to the default setting. |
| **query-max-response-time** <*seconds*> | Specifies the maximum response time (in seconds) advertised by this interface in queries when using IGMP V2. Hosts are allowed a random time within this period to respond, reducing response bursts. Use the **no** form of this command to return to the default setting. |

### Syntax Description

| | |
|---|---|
| **static-group** <*address*> | Configures the router's interface to be a statically-connected member of the specified group. Packets received on the correct RPF interface are forwarded to this interface regardless of whether any receivers have joined the specified group using IGMP. Use the **no** form of this command to remove a configured static group. |
| **version [1 \| 2]** | Sets the interface's IGMP version. Use the **no** form of this command to return to the default setting. |

### Default Values

| | |
|---|---|
| **ip igmp immediate-leave** | No default |
| **ip igmp last-member-query-interval** | 1000 milliseconds |
| **ip igmp querier-timeout** | 2x the **query-interval** value |
| **ip igmp query-interval** | 60 seconds |
| **ip igmp query-max-response-time** | 10 seconds |
| **ip igmp static-group** | No default |
| **ip igmp version** | Version 1 |

### Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |

### Usage Examples

The following example sets the query message interval on the interface to 200 milliseconds:

```
(config)#interface ethernet 0/1
(config-eth 0/1)#ip igmp last-member-query-interval 200
```

# ip mcast-stub downstream

Use the **ip mcast-stub downstream** command to enable multicast forwarding and IGMP (router mode) on an interface, and to place it in multicast stub downstream mode. Use the **no** form of this command to disable.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |

## Functional Notes

This command is used in IP multicast stub applications in conjunction with the **ip mcast-stub helper-address** and **ip mcast-stub upstream** commands. Downstream interfaces connect to segments with multicast hosts. Multiple interfaces may be configured in downstream mode; however, interfaces connecting to the multicast network (upstream) should not be configured in downstream mode. Interfaces configured as downstream should have the lowest IP address of all IGMP-capable routers on the connected segment in order to be selected as the designated router and ensure proper forwarding. Refer to *ip mcast-stub helper-address <ip address>* on page 546 and *ip mcast-stub upstream* on page 851 for more information.

## Usage Examples

The following example enables multicast forwarding and IGMP on the interface:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#ip mcast-stub downstream**

# ip mcast-stub fixed

Use the **ip mcast-stub fixed** command to allow forwarding of multicast traffic on a selected interface after enabling multicast routing. Use the **no** form of this command to disable this mode.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Functional Notes

Multicast routing must be enabled prior to setting **ip mcast-stub fixed** on the selected interface. Also, use the **ip igmp static-group** <*A.B.C.D*> command to receive multicast traffic without host-initiated Internet Group Management Protocol (IGMP) activity on the selected interface. Otherwise, all host-initiated IGMP transactions will enter multicast routes on the router's interface involved with IGMP activities.

### Usage Examples

The following example enables multicast traffic forwarding and IGMP on the interface:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#ip mcast-stub fixed**

# ip mcast-stub helper-enable

Use the **ip mcast-stub helper-enable** command to assign the **ip mcast-stub helper-address** as the IGMP proxy. Use the **no** form of this command to disable.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Functional Notes

This command is used in IP multicast stub applications in conjunction with the **ip mcast-stub helper-address**, **ip mcast-stub upstream**, and **ip mcast-stub downstream** commands. When enabled, the interface becomes a helper forwarding interface. The IGMP host function is dynamically enabled and the interface becomes the active upstream interface, enabling the unit to perform as an IGMP proxy. Refer to *ip mcast-stub helper-address <ip address>* on page 546, *ip mcast-stub downstream* on page 848, and *ip mcast-stub upstream* on page 851 for more information.

## Usage Examples

The following example sets the helper address as the IGMP proxy:

(config)**#interface ethernet 0/1**
(config-loop 1)**#ip mcast-stub helper-enable**

# ip mcast-stub upstream

Use the **ip mcast-stub upstream** command to enable multicast forwarding on an interface and place it in multicast stub upstream mode. Use the **no** form of this command to disable.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |

## Functional Notes

This command is used in IP multicast stub applications in conjunction with the **ip mcast-stub helper-address** and **ip mcast-stub downstream** commands. When enabled, the interface becomes a candidate to be a helper forwarding interface. If chosen as the best path toward the helper address by the router's unicast route table, the IGMP host function is dynamically enabled and the interface becomes the active upstream interface, enabling the router to perform as an IGMP proxy. Though multiple interfaces may be candidates, no more than one interface will actively serve as the helper forwarding interface. Refer to *ip mcast-stub helper-address <ip address>* on page 546 and *ip mcast-stub downstream* on page 848 for more information.

## Usage Examples

The following example enables multicast forwarding on the interface:

```
(config)#interface ethernet 0/1
(config-eth 0/1)#ip mcast-stub upstream
```

# ip ospf

Use the **ip ospf** command to customize OSPF settings (if needed). Use the **no** form of these commands to return to the default settings. Variations of this command include:

**ip ospf authentication-key** <*password*>
**ip ospf cost** <*value*>
**ip ospf dead-interval** <*seconds*>
**ip ospf hello-interval** <*seconds*>
**ip ospf message-digest-key [1 | 2] md5** <*key*>
**ip ospf priority** <*value*>
**ip ospf retransmit-interval** <*seconds*>
**ip ospf transmit-delay** <*seconds*>

## Syntax Description

| | |
|---|---|
| **authentication-key** <*password*> | Assigns a simple-text authentication password to be used by other routers using the OSPF simple password authentication. |
| **cost** <*value*> | Specifies the OSPF cost of sending a packet on the interface. This value overrides any computed cost value. Range is 1 to 65,535. |
| **dead-interval** <*seconds*> | Sets the maximum interval (in seconds) allowed between hello packets. If the maximum is exceeded, neighboring devices will determine that the device is down. Range is 0 to 32,767 seconds. |
| **hello-interval** <*seconds*> | Specifies the interval (in seconds) between hello packets sent on the interface. Range is 0 to 32,767 seconds. |
| **message-digest-key [1 \| 2] md5** <*key*> | Configures OSPF Message Digest 5 (MD5) authentication (16-byte max) keys. |
| **priority** <*value*> | Sets the OSPF priority. The value set in this field helps determine the designated router for this network. Range is 0 to 255. |
| **retransmit-interval** <*seconds*> | Specifies the interval (in seconds) between link-state advertisements (LSAs). Range is 0 to 32,767 seconds. |
| **transmit-delay** <*seconds*> | Sets the estimated time (in seconds) required to send an LSA on the interface. Range is 0 to 32,767 seconds. |

## Default Values

| | |
|---|---|
| **dead-interval** <*seconds*> | 40 seconds |
| **hello-interval** <*seconds*> | 10 seconds: Ethernet, point-to-point, Frame Relay, and PPP |
| **retransmit-interval** <*seconds*> | 5 seconds |
| **transmit-delay** <*seconds*> | 1 second |

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include Ethernet sub-interfaces. |

# ip ospf authentication

Use the **ip ospf authentication** command to authenticate an interface that is performing OSPF authentication. Use the **no** form of this command to return to the default setting. Variations of this command include:

**ip ospf authentication**
**ip ospf authentication message-digest**
**ip ospf authentication null**

## Syntax Description

| | |
|---|---|
| **message-digest** | Optional. Selects message-digest authentication type. |
| **null** | Optional. Specifies that no authentication is used. |

## Default Values

By default, this is set to null (meaning no authentication is used).

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include Ethernet sub-interfaces. |

## Usage Examples

The following example specifies that no authentication will be used on the Ethernet interface:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#ip ospf authentication null**

# ip ospf network

Use the **ip ospf network** command to specify the type of network on this interface. Use the **no** form of this command to return to the default setting. Variations of this command include:

**ip ospf network broadcast**
**ip ospf network point-to-point**

## Syntax Description

| | |
|---|---|
| **broadcast** | Sets the network type for broadcast. |
| **point-to-point** | Sets the network type for point-to-point. |

## Default Values

By default, Ethernet defaults to broadcast. PPP and Frame Relay default to point-to-point.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include Ethernet sub-interfaces |

## Functional Notes

A point-to-point network will not elect designated routers.

## Usage Examples

The following example designates a broadcast network type:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#ip ospf network broadcast**

# ip pim sparse-mode

Use the **ip pim sparse-mode** command to enable protocol-independent multicast (PIM) sparse mode for this interface. Use the **no** form of this command to disable PIM sparse mode.

## Syntax Description

No subcommands.

## Default Values

By default, PIM sparse mode for this interface is disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Functional Notes

PIM Sparse Mode is a multicast routing protocol that makes use of the unicast forwarding table. It builds unidirectional shared trees rooted at a Rendezvous Point (RP) for a multicast group or a shortest-path tree rooted at a specific source for a multicast group.

## Usage Examples

The following example enables PIM sparse mode on the interface:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#ip pim sparse-mode**

# ip pim-sparse dr-priority <*value*>

Use the **ip pim-sparse dr-priority** command to specify the priority for the designated router (DR). This command modifies the routers priority in the DR election process. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the priority of this interface (to be used when determining the DR). Valid range is 1 to 4,294,967,295. |

## Default Values

By default, the priority of all PIM interfaces is 1.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Functional Notes

Interfaces advertise their configured priority values in the hello messages transmitted on the interface. Routers use the priority values to determine the appropriate DR. The router on the network segment with the highest priority is selected as the DR. If a hello message is received on the interface from a router on the network segment and it does not contain a priority, the entire network segment defaults to DR selection based on IP addresses instead of priority. In this instance, the DR is selected as the router on the network segment that has the highest IP address. AOS will always include a priority in all transmitted hello messages. If no priority is specifically designated by the user, the priority is set as the default of 1.

## Usage Examples

The following example specifies a priority of **100** on the Ethernet 0/1 interface:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#ip pim-sparse dr-priority 100**

# ip pim-sparse hello-timer <*value*>

Use the **ip pim-sparse hello-timer** command to specify protocol-independent multicast (PIM) sparse hello timer period. This is the time interval at which periodic hellos are sent out on all interfaces of a PIM-capable router. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the interval (in seconds) at which periodic hellos are sent out of the interface. Valid range is 10 to 3600 seconds. |

## Default Values

By default, the hellos are transmitted on PIM interfaces every 60 seconds.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Functional Notes

Hello messages are used to inform neighbors of a router's presence. Hello messages normally generate a small amount of traffic on an interface. Setting the **hello-timer** to a small interval increases the number of hellos sent (thus increasing the amount of traffic). Set the **hello-timer** to a reasonable value, taking into consideration the bandwidth available on the interface.

## Usage Examples

The following example specifies hellos be sent on the Ethernet 0/1 interface every **3600** seconds:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#ip pim-sparse hello-timer 3600**

# ip pim-sparse nbr-timeout <*value*>

Use the **ip pim-sparse nbr-timeout** command to specify protocol-independent multicast (PIM) sparse neighbor timeout. This is the time interval after which a PIM-capable router will consider a neighbor not present. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the time interval in seconds after which a neighbor is considered not present. Valid range is 30 to 10,800 seconds. |

## Default Values

By default, the nbr-timeout is set to 105 seconds.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example sets the nbr-timeout to **300** seconds:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#ip pim-sparse nbr-timeout 300**

# ip pim-sparse override-interval <*value*>

Use the **ip pim-sparse override-interval** command to specify the protocol-independent multicast (PIM) sparse join/prune override interval. This delay interval is the period after a join/prune that another router on the LAN may override a join/prune. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the delay time in milliseconds. Valid range is 0 to 65,535 milliseconds. |

## Default Values

By default, the override interval is set to 2500 milliseconds.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example sets the override interval to **3000** milliseconds:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#ip pim-sparse override-interval 3000**

# ip pim-sparse propagation-delay <*value*>

Use the **ip pim-sparse propagation-delay** command to specify the expected propagation delay for join/prune messages. Set the propagation delay (in milliseconds) to estimate the amount of delay found in the local link. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the expected propagation delay in the local link in milliseconds. Valid range is 0 to 32,767 milliseconds. |

## Default Values

By default, the propagation delay is set to 500 milliseconds.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example sets the propagation delay to **300** milliseconds:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#ip pim-sparse propagation-delay 300**

# ip policy route-map *<name>*

Use the **ip policy route-map** command to assign a policy route-map to this interface. Use the **no** form of this command to remove the route-map policy.

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies the name of the policy route map to assign to this interface. |

## Default Values

By default, no policy route map is assigned to this interface.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example assigns the policy route map **policy1** to the interface:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#ip policy route-map policy1**

# ip proxy-arp

Use the **ip proxy-arp** command to enable proxy Address Resolution Protocol (ARP) on the interface. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, proxy ARP is enabled.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include Ethernet sub-interfaces. |

## Functional Notes

In general, the principle of proxy-arp allows a router to insert its IP address in the source IP address field of a packet (if the packet is from a host on one of its subnetworks). This allows hosts to reach devices on other subnetworks without implementing routing or specifying a default gateway.

If proxy ARP is enabled, the AOS will respond to all arp requests with its specified MAC address and forward packets accordingly.

Enabling proxy-arp on an interface may introduce unnecessary ARP traffic on the network.

## Usage Examples

The following enables proxy ARP on the Ethernet interface:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#ip proxy-arp**

# ip rip receive version

Use the **ip rip receive version** command to configure the RIP version the unit accepts in all RIP packets received on the interface. Variations of this command include:

**ip rip receive version 1**
**ip rip receive version 2**

## Syntax Description

| | |
|---|---|
| **1** | Accepts only RIP version 1 packets received on the interface. |
| **2** | Accepts only RIP version 2 packets received on the interface. |

## Default Values

By default, all interfaces implement RIP version 1 (the default value for the **version** command).

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include Ethernet sub-interfaces. |

## Functional Notes

Use the **ip rip receive version** command to specify a RIP version that overrides the **version** (in the Router RIP) configuration. Refer to *version* on page 1734 for more information.

The AOS only accepts one version (either **1** or **2**) on a given interface.

## Usage Examples

The following example configures the Ethernet interface to accept only RIP version 2 packets:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#ip rip receive version 2**

# ip rip send version

Use the **ip rip send version** command to configure the RIP version the unit sends in all RIP packets transmitted on the interface. Variations of this command include:

**ip rip send version 1**
**ip rip send version 2**

## Syntax Description

| | |
|---|---|
| **1** | Transmits only RIP version 1 packets on the interface. |
| **2** | Transmits only RIP version 2 packets on the interface. |

## Default Values

By default, all interfaces transmit RIP version 1 (the default value for the **version** command).

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include Ethernet sub-interfaces. |

## Functional Notes

Use the **ip rip send version** command to specify a RIP version that overrides the **version** (in the Router RIP) configuration. Refer to *version* on page 1734 for more information.

The AOS only transmits one version (either **1** or **2**) on a given interface.

## Usage Examples

The following example configures the Ethernet interface to transmit only RIP version 2 packets:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#ip rip send version 2**

# ip rip summary-address *<ip address> <subnet mask>*

Use the **ip rip summary-address** command to manually summarize the routes Routing Information Protocol (RIP) will advertise and send out a specified interface. Use the **no** form of this command to disable this mode.

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies the summarized network IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.0**). |
| *<subnet mask>* | Specifies the subnet mask that corresponds to the range of IP addresses (network). Subnet masks can be expressed in dotted decimal notation (for example, **255.255.255.0**) or as a prefix length (for example, **/24**). |

## Default Values

By default, no manual summarization is applied by RIP.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Functional Notes

Unlike the automatic summarization on classful network boundaries, only specific network advertisements are made by RIP using the **ip rip summary-address** command. This command is only effective if RIP version 2 is configured.

## Usage Examples

The following example enables manual summarization on the specified IP address:

(config)**#interface eth 0/1**
(config-eth 0/1)**#ip rip summary-address 10.10.123.0 255.255.255.0**

# ip route-cache

Use the **ip route-cache** command to enable fast-cache switching on the interface. Use the **no** form of this command to disable fast-cache switching and return to process switching mode.

*Using Network Address Translation (NAT) or the AOS firewall capabilities on an interface requires process switching mode (using the **no ip route-cache** command).*

## Syntax Description

No subcommands.

## Default Values

By default, fast-cache switching is enabled on all Ethernet and virtual Frame Relay sub-interfaces. IP route-cache is enabled for all virtual PPP interfaces.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include Ethernet sub-interfaces. |

## Functional Notes

Fast switching allows an IP interface to provide optimum performance when processing IP traffic.

## Usage Examples

The following example enables fast switching on the Ethernet interface:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#ip route-cache**

# ip unnumbered *<interface>*

Use the **ip unnumbered** command to use the IP address assigned to the specified interface for all IP processing on the active interface. Use the **no** form of this command to remove the unnumbered configuration.

## Syntax Description

| | |
|---|---|
| *<interface>* | Specifies the interface that contains the IP address to use as the source address for all packets transmitted on this interface. Specify an interface in the format *<interface type [slot/port \| slot/port.sub-interface id \| interface id \| interface id.sub-interface id]>*. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **show ip unnumbered ?** for a list of valid interfaces. |

## Default Values

By default, all interfaces are configured to use a specified IP address (using the **ip address** command).

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include Ethernet sub-interfaces. |
| Release 6.1 | Command was expanded to include VLAN interfaces. |
| Release 11.1 | Command was expanded to include demand interfaces. |

## Functional Notes

If **ip unnumbered** is enabled on an interface, all IP traffic from the interface will use a source IP address taken from the specified interface. For example, specifying **ip unnumbered ppp 1** while in the Ethernet Interface Configuration mode configures the Ethernet interface to use the IP address assigned to the PPP interface for all IP processing. In addition, the AOS uses the specified interface information when sending route updates over the unnumbered interface.

## Usage Examples

The following example configures the Ethernet interface 0/1 to use the IP address assigned to the PPP interface (**ppp 1**):

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#ip unnumbered ppp 1**

# ip urlfilter *<name>*

Use the **ip urlfilter** command to apply a URL filter to the interface for all inbound or outbound traffic. Use the **no** form of this command to remove the URL filter from an interface. Variations of this command include:

**ip urlfilter** <*name*> **in**
**ip urlfilter** <*name*> **out**

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies the URL filter name to use on the interface. |
| **in** | Applies the filter to the inbound traffic. |
| **out** | Applies the filter to the outbound traffic. |

## Default Values

By default, there are no URL filters applied to any interfaces.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Functional Notes

The firewall must be enabled using the **ip firewall** command in order to use URL filters. The URL filter must be created by using the **ip urlfilter *<filtername>* http** command before applying it to the interface. See *ip urlfilter <name> http* on page 598 for more information on using this command.

## Usage Example

The following example performs URL filtering on all traffic entering through the Ethernet interface and matches the URL filter named **MyFilter**:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#ip urlfilter MyFilter in**

# lldp receive

Use the **lldp receive** command to allow LLDP packets to be received on this interface. Use the **no** form of this command to prevent LLDP packets from being received on the interface.

## Syntax Description

No subcommands.

## Default Values

By default, all interfaces are configured to send and receive LLDP packets.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Usage Examples

The following example configures Ethernet interface 0/1 to receive LLDP packets:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#lldp receive**

# lldp send

Use the **lldp send** command to configure this interface to transmit LLDP packets or to control the types of information contained in the LLDP packets transmitted by this interface. Use the **no** form of these commands to prevent certain information from being transmitted by the interface. Variations of this command include:

**lldp send management-address**
**lldp send port-description**
**lldp send system-capabilities**
**lldp send system-description**
**lldp send system-name**
**lldp send-and-receive**

## Syntax Description

| | |
|---|---|
| **management-address** | Enables transmission of management address information on this interface. |
| **port-description** | Enables transmission of port description information on this interface. |
| **system-capabilities** | Enables transmission of this device's system capabilities on this interface. |
| **system-description** | Enables transmission of this device's system description on this interface. |
| **system-name** | Enables transmission of this device's system name on this interface. |
| **and-receive** | Configures this interface to both transmit and receive LLDP packets. This is the default setting. |

## Default Values

Be default, all interfaces are configured to transmit and receive LLDP packets of all types.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Functional Notes

Individual LLDP information can be enabled or disabled using the various forms of the **lldp send** command. For example, use the **lldp send and-receive** command to enable transmit and receive of all LLDP information. Then use the **no lldp send port-description** command to prevent LLDP from transmitting port description information.

## Usage Examples

The following example configures Ethernet interface 0/1 to transmit LLDP packets containing all enabled information types:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#lldp send**

The following example configures Ethernet interface 0/1 to transmit and receive LLDP packets containing all information types:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#lldp send-and-receive**

# mac-address *<mac address>*

Use the **mac-address** command to specify the Media Access Control (MAC) address of the unit. Only the last three values of the MAC address can be modified. The first three values contain the ADTRAN reserved number (00:0A:C8) by default. Use the **no** form of this command to return to the default MAC address programmed by ADTRAN.

## Syntax Description

| | |
|---|---|
| *<mac address>* | Specifies a valid 48-bit MAC address. MAC addresses should be expressed in following format xx:xx:xx:xx:xx:xx (for example, **00:A0:C8:00:00:01**). |

## Default Values

A unique default MAC address is programmed in each unit shipped by ADTRAN.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include Gigabit Ethernet interfaces. |

## Usage Examples

The following example configures a MAC address of **00:0A:C8:5F:00:D2**:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#mac-address 00:0A:C8:5F:00:D2**

# max-reserved-bandwidth <*value*>

Use the **max-reserved-bandwidth** command to specify the percentage of interface bandwidth reserved for use in user-defined (priority or class-based) queues. The remainder of the interface bandwidth is reserved for system critical traffic and is not available to user-defined queues. Use the **no** form of this command to restore the default values.


*Reserving a portion of the interface bandwidth for system critical traffic is necessary for proper operation. Specifying the entire interface bandwidth for use in user-defined queues can cause undesirable operation.*

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the maximum percentage of bandwidth to reserve for QoS. This setting is configured as a percentage of the total interface speed. Range is 1 to 100 percent. |

## Default Values

By default, **max-reserved-bandwidth** is set to 75 percent, which reserves 25 percent of the interface bandwidth for system critical traffic.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example specifies **85** percent of the bandwidth on the Ethernet interface 0/1 be available for use in user-defined queues:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#max-reserved-bandwidth 85**

# media-gateway ip

Use the **media-gateway ip** command to associate an IP address source to use for RTP traffic. When configuring VoIP, RTP traffic needs an IP address to be associated with it. However, some interfaces allow "dynamic" configuration of IP addresses, and thus, this value could change periodically. Use the **no** form of these commands to disable these functions. Variations of this command include:

**media-gateway ip loopback** <*ip address*>
**media-gateway ip primary**
**media-gateway ip secondary** <*ip address*>

## Syntax Description

| | |
|---|---|
| **loopback** <*ip address*> | Use an IP address statically defined to a loopback interface. Helpful when using a single IP address across multiple WAN interfaces for RTP traffic. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| **primary** | Use the IP address that is configured as primary on this interface. Applies to static, DHCP, or negotiated addresses. |
| **secondary** <*ip address*> | Use the statically defined secondary IP address of this interface to be used for RTP traffic. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

## Default Values

By default, **media-gateway ip** is disabled.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example configures the unit to use the primary IP address for RTP traffic:

(config)**#interface ethernet 0/1**
(config)**#media-gateway ip primary**

# mtu <*size*>

Use the **mtu** command to configure the maximum transmit unit (MTU) size for the active interface. Use the **no** form of this command to return to the default value.

## Syntax Description

| | | |
|---|---|---|
| <*size*> | Configures the window size for transmitted packets. The valid ranges for the various interfaces are listed below: | |
| | ATM interfaces | 64 to 1520 |
| | Demand interfaces | 64 to 1520 |
| | Ethernet interfaces | 64 to 1500 |
| | HDLC interfaces | 64 to 1520 |
| | Loopback interfaces | 64 to 1500 |
| | Tunnel interfaces | 64 to 18,190 |
| | Virtual Frame Relay sub-interfaces | 64 to 1520 |
| | Virtual PPP interfaces | 64 to 1500 |

## Default Values

| | | |
|---|---|---|
| <*size*> | The default values for the various interfaces are listed below: | |
| | ATM interfaces | 1500 |
| | Demand interfaces | 1500 |
| | Ethernet interfaces | 1500 |
| | HDLC interfaces | 1500 |
| | Loopback interfaces | 1500 |
| | Tunnel interfaces | 1500 |
| | Virtual Frame Relay sub-interfaces | 1500 |
| | Virtual PPP interfaces | 1500 |

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

OSPF will not become adjacent on links where the MTU sizes do not match. If router A and router B are exchanging hello packets but their MTU sizes do not match, they will never reach adjacency. This is by design and required by the RFC.

## Usage Examples

The following example specifies an MTU of **1200** on the Ethernet interface:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#mtu 1200**

# port-auth auth-mode

Use the **port-auth auth-mode** command to configure the authentication mode. Use the **no** form of this command to return to the default settings. Variations of this command include:

**port-auth auth-mode mac-based**
**port-auth auth-mode port-based**

## Syntax Description

| | |
|---|---|
| **mac-based** | Specifies a MAC-based authentication mode. Each host must authenticate separately. |
| **port-based** | Specifies a port-based authentication mode. Only a single host can participate in the authentication process. |

## Default Values

By default, the authentication mode is port-based.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example configures the unit for MAC-based authentication mode:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#port-auth auth-mode mac-based**

# port-auth control-direction

Use the **port-auth control direction** command to configure the direction in which traffic is blocked. This command is only applicable when authentication is port-based. Use the **no** form of this command to return to the default settings. Variations of this command include:

**port-auth control-direction both**
**port-auth control-direction in**

## Syntax Description

| | |
|---|---|
| **both** | Blocks traffic in both directions when the port becomes unauthorized. |
| **in** | Blocks only incoming traffic when the port becomes unauthorized. |

## Default Values

By default, traffic is blocked in both directions.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example causes traffic to be blocked in both directions when the port becomes unauthorized:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#port-auth control-direction both**

# port-auth multiple-hosts

Use the **port auth multiple-hosts** command to allow multiple hosts to access an authorized port without going through the authentication process. This command is only applicable when authentication is port-based. Use the **no** form of this command to return to the default settings.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example enables multiple hosts to access an authorized port:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#port-auth multiple-hosts**

# port-auth port-control

Use the **port-auth port-control** command to configure the port-authorization state. Use the **no** form of this command to return to the default settings. Variations of this command include:

**port-auth port-control auto**
**port-auth port-control force-authorized**
**port-auth port-control force-unauthorized**

## Syntax Description

| | |
|---|---|
| **auto** | Enables the port-authentication process. |
| **force-authorized** | Forces the port into an authorized state. |
| **force-unauthorized** | Forces the port into an unauthorized state. |

## Default Values

By default, all ports are forced to an authorized state.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example forces Ethernet port 0/1 into an unauthorized state:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#port-auth port-control force unauthorized**

# power inline

Use the **power inline** command to detect attached Powered Devices (PDs) and deliver 48 VDC, compliant with the IEEE 802.3af power-over-Ethernet standard, to the PD via existing CAT5 cabling. To disable power detection and supply, use the **power inline never** command. Variations of this command include:

**power inline auto**
**power inline legacy**
**power inline never**

## Syntax Description

| | |
|---|---|
| **auto** | Enables power detection and supply to PDs. |
| **legacy** | Enables power detection and supply of legacy non-IEEE 802.3af compliant PDs. |
| **never** | Disables power detection and supply to PDs. |

## Default Values

By default, PWR switches discover and provide power to IEEE compliant PDs.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Usage Examples

The following example configures the ethernet interface to detect and supply power to PDs:

(config)**#interface ethernet 0/3**
(config-eth 0/3)**#power inline auto**

# qos-policy out *<name>*

Use the **qos-policy out** command to apply a previously-configured QoS map to outgoing packets on an interface. Use the **no** form of this command to remove the map from the interface.

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies the name of a previously-created QoS map (refer to *qos map <name> <number>* on page 643 for more information). |

## Default Values

No default value is necessary for this command.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Functional Notes

When a QoS policy is applied to an interface, it may be disabled if the interface bandwidth is not adequate to support the requested bandwidth on the map set. Once the bandwidth problem is resolved, the map will work again. The bandwidth will be rechecked on any of the following changes:

1. A priority or class-based entry is added to, deleted from, or changed in a QoS map set.
2. The interface bandwidth is changed by the **bandwidth** command on the interface.
3. A QoS policy is applied to an interface.
4. A cross-connect is created that includes an interface with a QoS policy.
5. The interface queuing method is changed to fair-queue to use weighted fair queuing.
6. The interface operational status changes.
7. The interface bandwidth changes for other reasons (e.g., when ADSL finishes training).

In order to prevent the map from being disabled in cases of temporary inadequate bandwidth (e.g., a single link goes down in a dual T1 multilink configuration where the map requests more than one T1's worth of bandwidth), the QoS map uses the maximum theoretical bandwidth on an interface, not the actual bandwidth at that time. This actually helps QoS keep higher priority class-based traffic working better than best-effort traffic when the bandwidth drops.

## Usage Examples

The following example applies the QoS map **VOICEMAP** to the Ethernet interface:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#qos-policy out VOICEMAP**

# qos

Use the **qos** (quality of service) command to set the interface to the trusted state and to set the default cost of service (CoS) value. To return to defaults, use the **no** version of this command. Variations of this command include:

**qos default-cos** <*value*>
**qos trust cos**

## Syntax Description

| | |
|---|---|
| **default-cos** <*value*> | Sets the default CoS value for untrusted ports and all untagged packets. Range is 0 through 7. |
| **trust cos** | Sets the interface to the trusted state. |

## Default Values

By default, the interface is untrusted with a default CoS of 0.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

Set the interface to **trust cos** if received 802.1P CoS values are considered valid (i.e., no need to reclassify) and do not need to be tagged with the default value. When set to untrusted, the **default-cos** value for the interface is used.

## Usage Examples

The following example sets Ethernet interface 0/1 as a trusted interface and assigns untagged packets a CoS value of **1**:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#qos trust cos**
(config-eth 0/1)**#qos default-cos 1**

# snmp trap

Use the **snmp trap** command to enable all supported Simple Network Management Protocol (SNMP) traps on the interface. Us the **no** form of this command to disable this trap.

## Syntax Description

No subcommands.

## Default Values

By default, all interfaces (except virtual Frame Relay interfaces and sub-interfaces) have SNMP traps enabled.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 3.1 | Command was extended to the SHDSL interface. |
| Release 5.1 | Command was expanded to include Ethernet sub-interfaces and Gigabit Ethernet interfaces. |

## Usage Examples

The following example enables SNMP capability on the Ethernet interface:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#snmp trap**

# snmp trap link-status

Use the **snmp trap link-status** command to control the SNMP variable ifLinkUpDownTrapEnable (RFC2863) to enable (or disable) the interface to send SNMP traps when there is an interface status change. Use the **no** form of this command to disable this trap.

## Syntax Description

No subcommands.

## Default Values

By default, the ifLinkUpDownTrapEnable OID is enabled for all interfaces except virtual Frame Relay interfaces.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 3.1 | Command was extended to the SHDSL interface. |
| Release 5.1 | Command was expanded to include Gigabit Ethernet, port channel, VLAN, E1, and G.703 interfaces. |

## Functional Notes

The **snmp trap link-status** command is used to control the RFC2863 ifLinkUpDownTrapEnable OID (OID number 1.3.6.1.2.1.31.1.1.1.14.0).

## Usage Examples

The following example disables the link-status trap on the interface:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#no snmp trap link-status**

# spanning-tree bpdufilter

Use the **spanning-tree bpdufilter** command to enable or disable the BPDU filter on a specific interface. This setting overrides the related Global setting (refer to *spanning-tree edgeport bpdufilter default* on page 676). Use the **no** version of the command to return to the default setting. Variations of this command include:

**spanning-tree bpdufilter disable**
**spanning-tree bpdufilter enable**

## Syntax Description

| | |
|---|---|
| **disable** | Disables BPDU filter for this interface. |
| **enable** | Enables BPDU filter for this interface. |

## Default Values

By default, this setting is disabled.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

The bpdufilter blocks any BPDUs from being transmitted and received on an interface.

## Usage Examples

The following example enables the BPDU filter on the Ethernet interface 0/3:

(config)**#interface ethernet 0/3**
(config-eth 0/3)**#spanning-tree bpdufilter enable**

The BPDU filter can be disabled on the Ethernet interface 0/3 by issuing the following commands:

(config)**#interface ethernet 0/3**
(config-eth 0/3)**#spanning-tree bpdufilter disable**

# spanning-tree bpduguard

Use the **spanning-tree bpduguard** command to enable or disable the BPDU guard on a specific interface. This setting overrides the related global setting (refer to *spanning-tree forward-time <value>* on page 679). Use the **no** version of the command to return to the default setting. Variations of this command include:

**spanning-tree bpduguard disable**
**spanning-tree bpduguard enable**

## Syntax Description

| | |
|---|---|
| **disable** | Disables BPDU guard for this interface. |
| **enable** | Enables BPDU guard for this interface. |

## Default Values

By default, this setting is disabled.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

The bpduguard blocks any BPDUs from being received on an interface.

## Usage Examples

The following example enables the BPDU guard on the interface Ethernet interface 0/3:

(config)**#interface ethernet 0/3**
(config-eth 0/3)**#spanning-tree bpduguard enable**

The BPDU guard can be disabled on the Ethernet interface 0/3 by issuing the following commands:

(config)**#interface ethernet 0/3**
(config-eth 0/3)**#spanning-tree bpduguard disable**

# spanning-tree cost <*value*>

Use the **spanning-tree cost** command to assign a cost to the interface. The cost value is used when computing the spanning-tree root path. Use the **no** version of the command to return to the default setting.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies a cost value of 1 to 200,000,000. |

## Default Values

By default, the cost value is set to 1000/link speed in Mbps.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following example sets the interface to a path cost of 1200:

(config)**#interface ethernet 0/3**
(config-eth 0/3)**#spanning-tree cost 1200**

# spanning-tree edgeport

Use the **spanning-tree edgeport** command to configure the interface to be an edgeport. This command overrides the related Global setting (refer to *spanning-tree edgeport default* on page 678). Use the **no** version of the command to return to the default setting.

## Syntax Description

No subcommands.

## Default Values

By default, this setting is disabled.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

Enabling this command configures the interface to go to a forwarding state when the link becomes active. When not enabled, an interface must go through the listening and learning states before going to the forwarding state.

## Usage Examples

The following example configures the interface to be an edgeport:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#spanning-tree edgeport**

An individual interface can be configured to not be considered an edgeport. For example:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#no spanning-tree edgeport**

# spanning-tree link-type

Use the **spanning-tree link-type** command to configure the spanning tree protocol link type for each interface. Use the **no** version of the command to return to the default setting. Variations of this command include:

**spanning-tree link-type auto**
**spanning-tree link-type point-to-point**
**spanning-tree link-type shared**

## Syntax Description

| | |
|---|---|
| **auto** | Determines link type by the port's duplex settings. |
| **point-to-point** | Manually sets link type to **point-to-point**, regardless of duplex settings. |
| **shared** | Manually sets link type to **shared**, regardless of duplex settings. |

## Default Values

By default, the interface is set to auto.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

This command overrides the default link type setting determined by the duplex of the individual port. By default, a port configured for half-duplex is set to **shared** link type, and a port configured for full-duplex is set to **point-to-point** link type. Setting the link type manually overrides the default and forces the port to use the specified link type. Use the **link-type auto** command to restore the convention of determining link type based on duplex settings.

## Usage Examples

The following example forces the link type to **point-to-point**, even if the port is configured to be half-duplex:

(config)**#interface ethernet 0/3**
(config-eth 0/3)**#spanning-tree link-type point-to-point**

## Technology Review

Rapid transitions are possible in rapid spanning-tree protocol (RSTP) by taking advantage of point-to-point links (a port is connected to exactly one other bridge) and edge-port connections (a port is not connected to any additional bridges). Setting the link-type to **auto** allows the spanning-tree to automatically configure the link type based on the duplex of the link. Setting the link type to **point-to-point** allows a half-duplex link to act as if it were a point-to-point link.

# spanning-tree pathcost method

Use the **spanning-tree pathcost** command to select a short or long method used by the spanning-tree protocol. Variations of this command include:

**spanning-tree pathcost method long**
**spanning-tree pathcost method short**

## Syntax Description

| | |
|---|---|
| **long** | Specifies 32-bit values when calculating pathcosts. |
| **short** | Specifies 16-bit values when calculating pathcosts. |

## Default Values

By default, **spanning-tree pathcost** is set to **short**.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following example specifies that the spanning tree protocol use a long pathcost method:

(config)**#interface ethernet 0/3**
(config-eth 0/3)**#spanning-tree pathcost method long**

# spanning-tree port-priority *<value>*

Use the **spanning-tree port-priority** command to select the priority level of this interface. To return to the default setting, use the **no** version of this command.

## Syntax Description

| | |
|---|---|
| *<value>* | Specifies a priority-level value from 0 to 240 (this value must be in increments of 16). |

## Default Values

By default, this set to 128.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

The only time that this priority level is used is when two interfaces with a path to the root have equal cost. At that point, the level set in this command will determine which port the spanning tree will use. Set the priority value lower to increase the chance the interface will be used.

## Usage Examples

The following example sets the interface to a priority of 100:

(config)**#interface ethernet 0/3**
(config-eth 0/3)**#spanning-tree port-priority 100**

# speed

Use the **speed** command to configure the speed of an Ethernet interface. Use the **no** form of this command to return to the default value. Variations of this command include:

**speed 10**
**speed 100**
**speed 1000**
**speed auto**

## Syntax Description

| | |
|---|---|
| **10** | Specifies 10 Mbps Ethernet. |
| **100** | Specifies 100 Mbps Ethernet. |
| **1000** | Specifies 1 Gbps Ethernet. This only applies to Gigabit Ethernet interfaces. |
| **auto** | Automatically detects 10 or 100 Mbps Ethernet and negotiates the duplex setting. |


*Some Ethernet equipment (though rare) is unable to negotiate duplex if speed is manually determined. To avoid incompatibilities, manually set the duplex if the speed is manually set. Refer to full-duplex* *on page 833* *and half-duplex* *on page 834**.*

## Default Values

By default, speed is set to **auto**.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example configures the Ethernet port for 100 Mb operation:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#speed 100**

# storm-control

Use the **storm-control** command to configure limits on the rates of broadcast, multicast, and unicast traffic on a port. Use the **no** version of this command to disable this feature. Variations of this command include:

**storm-control broadcast level** *<rising level>*
**storm-control broadcast level** *<rising level> <falling level>*
**storm-control multicast level** *<rising level>*
**storm-control multicast level** *<rising level> <falling level>*
**storm-control unicast level** *<rising level>*
**storm-control unicast level** *<rising level> <falling level>*

## Syntax Description

| | |
|---|---|
| **broadcast level** | Sets levels for broadcast traffic. |
| **multicast level** | Sets levels for multicast traffic. |
| **unicast level** | Sets levels for unicast traffic. |
| *<rising level>* | Specifies a rising level which determines the percentage of total bandwidth the port accepts before it begins blocking packets. Range is 1 to 100 percent. |
| *<falling level>* | Optional. Specifies a falling level which determines when the storm is considered over, causing the AOS to no longer block packets. This level must be less than the rising level. Range is 1 to 100 percent. |

## Default Values

By default, **storm-control** is disabled.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

This setting configures the rising and falling suppression values. When the selected rising level (which is a percentage of total bandwidth) is reached, the port begins blocking packets of the specified type (i.e., broadcast, multicast, or unicast). The AOS uses the rising level as its falling level if no falling level is specified.

## Usage Examples

The following example sets the rising suppression level to **85** percent for multicast packets:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#storm-control multicast level 85**

The following example sets the rising suppression level to **80** percent for broadcast packets, with a falling level of **50** percent:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#storm-control broadcast level 80 50**

# storm-control action shutdown

Use the **storm-control action shutdown** command shuts down the interface when a storm occurs. Use the **no** version of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled; the interface will only filter traffic.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

Enabling this option shuts down the interface if a multicast, unicast, or broadcast storm occurs.

## Usage Examples

The following example shuts down Ethernet interface 0/1 if a storm is detected:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#storm-control action shutdown**

# switchport access vlan *<vlan id>*

Use the **switchport access vlan** command to set the port to be a member of the VLAN when in access mode. To reset the port to be a member of the default VLAN, use the **no** version of this command.

## Syntax Description

| | |
|---|---|
| *<vlan id>* | Specifies a valid VLAN interface ID. Range is 1 to 4094. |

## Default Values

By default, this is set to VLAN 1 (the default VLAN).

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

If the port is in the trunk mode, this command will not alter the switchport mode to access. Instead it will save the value to be applied when the port does switch to access mode. Refer to *switchport mode* on page 899 for more information.

## Usage Examples

The following example sets the switchport mode to static access and makes the Ethernet interface 0/1 port a member of VLAN 2:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#switchport mode access**
(config-eth 0/1)**#switchport access vlan 2**

# switchport gvrp

Use the **switchport gvrp** command to enable or disable GARP VLAN Registration Protocol (GVRP) on an interface. Use the **no** form of this command to return to the default value.

## Syntax Description

No subcommands.

## Default Values

By default, GVRP is disabled on all ports.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Functional Notes

Enabling GVRP on any interface enables GVRP globally.

## Usage Examples

The following example enables GVRP on Ethernet interface 0/24:

(config)**#interface ethernet 0/24**
(config-eth 0/24)**#switchport gvrp**

# switchport mode

Use the **switchport mode** command to configure the VLAN membership mode. To reset membership mode to the default value, use the **no** version of this command. The **stack** selection does not apply to the NetVanta 300 Series units. Variations of this command include:

**switchport mode access**
**switchport mode stack**
**switchport mode trunk**

## Syntax Description

| | |
|---|---|
| **access** | Sets port to be a single (non-trunked) port that transmits and receives no tagged packets. |
| **stack** | Sets the port to allow it to communicate with a switch stack. (Does not apply the NetVanta 300 Series units.) |
| **trunk** | Sets port to transmit and receive packets on all VLANs included within its VLAN allowed list. |

## Default Values

By default, **switchport mode** is set to **access**.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

Configuring the interface for stack mode (using the **switchport mode stack** command) enables the switch to communicate with other switches that it is stacking capable.

- If the switch is configured as the stack master (using the (config)#stack master command), it will begin advertising itself as a stack master.
- If the switch is configured as the stack member (using the (config)#stack member command), it will begin advertising other stack masters that it knows about.

Stack mode also allows the port to transmit and receive packets on all VLANs that are included in the VLAN allowed list.

## Usage Examples

The following example sets the port to be a trunk port:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#switchport mode trunk**

# switchport protected

Use the **switchport protected** command to prevent the port from transmitting traffic to all other protected ports. A protected port can only send traffic to unprotected ports. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

This command is disabled by default.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

In the example below, all three of the ports are on VLAN 3, and Ethernet 0/1 and Ethernet 0/2 are designated as protected ports. Ethernet 0/3 is unprotected. Ethernet 0/1 and Ethernet 0/2 will be allowed to send traffic to Ethernet 0/3, but traffic traveling between Ethernet 0/1 and Ethernet 0/2 will be blocked.

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#switchport access vlan 3**
(config-eth 0/1)**#switchport protected**
(config-eth 0/1)**#exit**

(config)**#interface ethernet 0/2**
(config-eth 0/2)**#switchport access vlan 32**
(config-eth 0/2)**#switchport protected**
(config-eth 0/1)**#exit**

(config)**#interface ethernet 0/3**
(config-eth 0/3)**#switchport access vlan 3**

# switchport port-security

Use the **switchport port-security** command to enable port security functionality on the interface. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

This command is disabled by default.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Functional Notes

You cannot enable port security on a port that is already configured as the following:

- Monitor session destination
- Member of a port channel interface
- Dynamic or trunk port (i.e., the port must be configured as static access)

## Usage Examples

The following example enables port security on the Ethernet interface 0/1 interface:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#switchport port-security**

# switchport port-security aging

Use the **switchport port-security aging** command to enable and configure secure MAC address aging on a particular interface. Use the **no** form of this command to disable this feature. Variations of this command include:

**switchport port-security aging static**
**switchport port-security aging time** <*value*>
**switchport port-security aging type absolute**

## Syntax Description

| | |
|---|---|
| **static** | Configures the interface to age static as well as dynamic entries in the secure MAC address table. |
| **time** <*value*> | Enables port security aging for dynamic entries in the secure MAC address table by configuring a time (in minutes). Disable aging by setting the time to 0. |
| **type absolute** | Configures the address to be removed after the specified time, regardless of activity. |

## Default Values

By default, dynamic and static aging are disabled.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Usage Examples

The following example sets the aging time of secure MAC addresses to 10 minutes:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#switchport port-security aging time 10**

# switchport port-security expire

Use the **switchport port-security expire** command to disable an interface after a specified amount of time. Use the **no** form of this command to return to the default setting. Variations of this command include:

**switchport port-security expire time** <*value*>
**switchport port-security expire type absolute**

## Syntax Description

| | |
|---|---|
| **time** <*value*> | Enables port expiration by configuring a time (in minutes). Disable by setting time to 0. |
| **type absolute** | Configures the interface to shut down after the specified time, regardless of activity. |

## Default Values

By default, this command is disabled and set to **type absolute**.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Usage Examples

The following example disables Ethernet interface 0/1 after 10 minutes:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#switchport port-security expire time 10**

# switchport port-security mac-address

Use the **switchport port-security mac-address** command to add a static secure MAC address or sticky secure MAC address associated with the interface and to enable sticky address learning. Variations of this command include the following: Us the **no** form of this command to remove a MAC address associated with this port.

**switchport port-security mac-address** <*mac address*>
**switchport port-security mac-address sticky**
**switchport port-security mac-address sticky** <*mac address*>

## Syntax Description

| | |
|---|---|
| **sticky** | Optional. Enables sticky address learning if no MAC address is specified. |
| <*mac address*> | Optional. Adds a MAC address associated with this interface. MAC addresses should be expressed in following format xx:xx:xx:xx:xx:xx (for example, **00:A0:C8:00:00:01**). |

## Default Values

By default, sticky learning is disabled and there are no configured MAC addresses.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Usage Examples

The following example adds a single static address and enables sticky address learning on interface Ethernet interface 0/1:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#switchport port-security mac-address 00:A0:C8:02:D0:30**
(config-eth 0/1)**#switchport port-security mac-address sticky**

# switchport port-security maximum <*value*>

Use the **switchport port-security maximum** command to configure the maximum number of secure MAC addresses associated with the interface. Use the **no** form of this command to return to the default setting.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the maximum number of secure MAC addresses to be associated with the interface. Range is 1 to 132. |

## Default Values

The default value for this command is 1.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Usage Examples

The following example sets the maximum supported MAC addresses for Ethernet interface 0/1 to **2**:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#switchport port-security maximum 2**

# switchport port-security violation

Use the **switchport port-security violation** command to configure the action to be taken once a security violation is encountered. Use the **no** form of this command to return to the default setting. Variations of this command include:

**switchport port-security violation protect**
**switchport port-security violation restrict**
**switchport port-security violation shutdown**

## Syntax Description

| | |
|---|---|
| **protect** | Determines that the unit will not learn any new secure addresses (nor allow these new sources to pass traffic) until the number of currently active secure addresses drops below the maximum setting. |
| **restrict** | Determines that the security violation counter increments and an SNMP trap is sent once a violation is detected. The new address is not learned and data from that address is not allowed to pass. |
| **shutdown** | Determines that the interface is disabled once a violation is detected. A **no shutdown** command is required to re-enable the interface. |

## Default Values

The default for this command is shutdown.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Usage Examples

The following example configures the interface to react to security violations by not learning the addresses of and not accepting data from the violation source:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#switchport port-security violation restrict**

# switchport trunk allowed vlan

Use the **switchport trunk allowed vlan** command to allow certain VLANs to transmit and receive traffic on this port when the interface is in trunking mode. To return to defaults, use the **no** version of this command. Variations of this command include:

**switchport trunk allowed vlan** <*list*>
**switchport trunk allowed vlan add** <*list*>
**switchport trunk allowed vlan all**
**switchport trunk allowed vlan except** <*list*>
**switchport trunk allowed vlan none**
**switchport trunk allowed vlan remove** <*list*>

## Syntax Description

| | |
|---|---|
| <*list*> | Specifies a list of valid VLAN interface IDs. Refer to *Functional Notes*, below. |
| **add** | Adds the specified VLAN IDs to the VLAN trunking allowed list. |
| **all** | Adds all configured VLAN IDs to the VLAN trunking allowed list. |
| **except** | Adds all configured VLAN IDs to the VLAN trunking allowed list except those specified in the <*vlan id list*>. |
| **none** | Adds no VLAN IDs to the VLAN trunking allowed list. |
| **remove** | Removes VLAN IDs from the VLAN trunking allowed list. |

## Default Values

By default, all valid VLANs are allowed.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

A VLAN list is a set of VLAN IDs delimited by commas. A valid VLAN ID value must be from 1 through 4094. A range of IDs may be expressed as a single element by using a hyphen between endpoints. For example the VLAN ID range **1,2,3,4,6,7,8,9,500** may be more easily expressed as **1-4,6-9,500**. No spaces are allowed in a valid ID range.

## Usage Examples

The following example adds VLANs to the previously existing list of VLANs allowed to transmit and receive on this port:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#switchport trunk allowed vlan add 1-4,7-9,500**

# switchport trunk fixed vlan

Use the **switchport trunk fixed vlan** command to change the configured list of VLANs that remain fixed in use only when GVRP is enabled on the interface. Of these VLANs, VLANs statically created will be available for use on the interface. Use the **no** forms of these commands to disable these features. Variations of this command include:

**switchport trunk fixed vlan add** <*list*>
**switchport trunk fixed vlan all**
**switchport trunk fixed vlan except** <*list*>
**switchport trunk fixed vlan none**
**switchport trunk fixed vlan remove** <*list*>

## Syntax Description

| | |
|---|---|
| <*list*> | Specifies a list of valid VLAN interface IDs. Refer to *Functional Notes*, below. |
| **add** | Adds VLANs to the VLAN GVRP trunking fixed list. |
| **all** | Adds all VLANs to the VLAN GVRP trunking fixed list. |
| **except** | Adds all VLAN IDs to the VLAN trunking fixed list except those in the command line VLAN ID list. |
| **none** | Removes all VLANs from the VLAN GVRP trunking fixed list. |
| **remove** | Removes VLAN from the VLAN trunking fixed list. |

## Default Values

By default, no VLANs are in the VLAN GVRP trunking fixed list (**switchport trunk fixed vlan none**).

A VLAN list is a set of VLAN IDs delimited by commas. A valid VLAN ID value must be from 1 through 4094. A range of IDs may be expressed as a single element by using a hyphen between endpoints. For example the VLAN ID range **1,2,3,4,6,7,8,9,500** may be more easily expressed as **1-4,6-9,500**. No spaces are allowed in a valid ID range.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Functional Notes

This command has no effect on VLAN membership configuration unless GVRP is enabled on the interface.

## Usage Examples

The following example changes the configured list of fixed VLANs by adding VLAN 50 to the list.

(config-eth 0/20)**#switchport trunk fixed vlan add 1-15,25-30,40**
(config-eth 0/20)**#switchport trunk fixed vlan add 50**

The following example changes the configured list of fixed VLANs by removing VLANs 10 to 100 from the list:

(config-eth 0/20)**#switchport trunk fixed vlan remove 10-100**

The following example changes the configured list of fixed VLANs to include only VLANs 1 to 1000:

(config-eth 0/20)**#switchport trunk fixed vlan 1-1000**

The following example changes the configured list of fixed VLANs to include no VLANs (except those VLANs that are native):

(config-eth 0/20)**#switchport trunk fixed vlan none**

# switchport trunk native vlan *<vlan id>*

Use the **switchport trunk native vlan** command to set the VLAN native to the interface when the interface is in trunking mode. To return to defaults, use the **no** version of this command.

## Syntax Description

| | |
|---|---|
| *<vlan id>* | Specifies a valid VLAN interface ID. Range is 1 to 4094. |

## Default Values

By default, this is set to VLAN 1.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

Configure which VLAN the interface uses as its native VLAN during trunking. Packets from this VLAN leaving the interface will not be tagged with the VLAN number. Any untagged packets received by the interface are considered a part of the native VLAN ID.

## Usage Examples

The following example sets the native VLAN on Ethernet interface 0/1 to VLAN 2:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#switchport trunk native vlan 2**

# traffic-shape rate <*value*>

Use the **traffic-shape rate** command to specify and enforce an output bandwidth for Ethernet and VLAN interfaces. Variations of this command include:

**traffic-shape rate** <*value*>
**traffic-shape rate** <*value*> <*burst*>

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the rate (in bits per second) at which the interface should be shaped. |
| <*burst*> | Optional. Specifies the allowed burst in bytes. By default, the burst is specified as the rate divided by 5 and represents the number of bytes that would flow within 200 ms. |

Default Values

By default, traffic-shaping rate is disabled.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Functional Notes

Traffic shaping can be used to limit an Ethernet segment to a particular rate or to specify use of QoS on Ethernet or VLAN interfaces.

## Usage Examples

The following example sets the outbound rate of Ethernet interface 0/1 to 128 kbps and applies a QoS policy that all RTP traffic is given priority over all other traffic:

(config)**#qos map voip 1**
(config-qos-map)**#match ip rtp 10000 10500 all**
(config-qos-map)**#priority unlimited**
(config-qos-map)**#interface ethernet 0/1**
(config-eth)**#traffic-shape rate 128000**
(config-eth)**#qos-policy out voip**

# **vlan-id** <*vlan id*>

Use the **vlan-id** command to set a VLAN ID for the Ethernet interface. Use the **no** form of this command to remove an entry. Variations of this command include:

**vlan-id** <*vlan id*>
**vlan-id** <*vlan id*> **native**

## Syntax Description

| | |
|---|---|
| <*vlan id*> | Specifies a valid VLAN interface ID number. Range is 1 to 4095. |
| **native** | Optional. Specifies that data for that VLAN ID goes out untagged. If **native** is not specified, data for that VLAN ID goes out tagged. |

## Default Values

By default, no VLAN ID is set.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Usage Examples

The following example configures a native VLAN of 5 for the Ethernet interface 0/1:

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#vlan-id 5 native**

# FDL INTERFACE CONFIGURATION COMMAND SET

FDL Interface Configuration mode is used for establishing a Telnet session over the FDL (facility datalink). To activate, enter the **interface fdl** command and specify the associated slot/port number (of the T1 interface used) at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#interface fdl 1/1
(config-fdl 1/1)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:



All other commands for this command set are described in this section in alphabetical order.



# ip address *<ip address> <subnet mask>*

Use the **ip address** command to define an IP address on the specified interface. Use the **no** form of this command to remove a configured IP address.

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies a valid IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| *<subnet mask>* | Specifies the subnet mask that corresponds to a range of IP addresses (network) or a specific host. Subnet masks can be expressed in dotted decimal notation (for example, **255.255.255.0**) or as a prefix length (for example, **/24**). |

## Default Values

By default, there are no assigned IP addresses.

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced for the FDL interface. |

## Usage Examples

The following example configures an IP address of **192.22.72.101/30**:

(config)**#interface fdl 1/1**
(config-fdl 1/1)**#ip address 192.22.72.101 255.255.255.252**

# ip learn-address

Use the **ip learn-address** command to automatically learn the IP address of the remote unit. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |

## Usage Examples

The following example enables the FDL to automatically learn the remote unit's IP address:

```
(config)#interface fdl 1/1
(config-fdl 1/1)#ip learn-address
```

# mtu <*size*>

Use the **mtu** command to configure the maximum transmit unit (MTU) size for the active interface. Use the **no** form of this command to return to the default value.

## Syntax Description

| | | |
|---|---|---|
| <*size*> | Configures the window size for transmitted packets. The valid ranges for the various interfaces are listed below: | |
| | ATM interfaces | 64 to 1520 |
| | Demand interfaces | 64 to 1520 |
| | Ethernet interfaces | 64 to 1500 |
| | FDL interfaces | 64 to 256 |
| | HDLC interfaces | 64 to 1520 |
| | Loopback interfaces | 64 to 1500 |
| | Tunnel interfaces | 64 to 18,190 |
| | Virtual Frame Relay sub-interfaces | 64 to 1520 |
| | Virtual PPP interfaces | 64 to 1500 |

## Default Values

| | | |
|---|---|---|
| <*size*> | The default values for the various interfaces are listed below: | |
| | ATM interfaces | 1500 |
| | Demand interfaces | 1500 |
| | Ethernet interfaces | 1500 |
| | FDL interfaces | 256 |
| | HDLC interfaces | 1500 |
| | Loopback interfaces | 1500 |
| | Tunnel interfaces | 1500 |
| | Virtual Frame Relay sub-interfaces | 1500 |
| | Virtual PPP interfaces | 1500 |

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 7.1 | Command expanded to support FDL. |

## Functional Notes

OSPF will not become adjacent on links where the MTU sizes do not match. If router A and router B are exchanging hello packets but their MTU sizes do not match, they will never reach adjacency. This is by design and required by the RFC.

## Usage Examples

The following example specifies an MTU of 128 on the FDL interface:

(config)**#interface fdl 1/1**
(config-fdl 1/1)**#mtu 128**

# peer default ip address *<ip address>*

Use the **peer default ip address** command to specify the default IP address of the remote end of this interface. Use the **no** form of this command to remove a default IP address.

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies the default IP address for the remote end. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

## Default Values

By default, there is no assigned peer default IP address.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |
| Release 7.1 | Command was expanded to include FDL. |

## Functional Notes

This command is useful if the peer does not send the IP address option during PPP negotiations.

## Usage Examples

The following example sets the default peer IP address to 192.22.71.50:

(config)**#interface fdl 1/1**
(config-fdl 1/1)**#peer default ip address 192.22.71.50**

# FXO INTERFACE CONFIGURATION COMMAND SET

To activate the FXO Interface Configuration mode, enter the **interface fxo** command and specify the FXO port at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#interface fxo 0/1
(config-fxo 0/1)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:



All other commands for this command set are described in this section in alphabetical order.



# impedance

Use the **impedance** command to configure the AC impedance of the 2-wire interface. Use the **no** form of this command to return to the default value. Variations of this command include:

**impedance 600c**
**impedance 900c**

## Syntax Description

| | |
|---|---|
| **600c** | Specifies an impedance of 600 Ω **+** 2.16 μF**.** |
| **900c** | Specifies an impedance of 900 Ω + 2.16 μF. |

## Default Value

By default the impedance is set to **600c**.

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |

## Usage Examples

The following example sets the impedance to 600 Ω + 2.16 μF:

(config)**#interface fxo 0/1**
(config-fxo 0/1)**#impedance 600c**

# loopback

Use the **loopback** command to activate a loopback on the FXO Module. Use the **no** form of this command to deactivate the loopback. Variations of this command include:

**loopback analog**
**loopback digital**

## Syntax Description

| | |
|---|---|
| **analog** | Initiates a loopback toward the T1 network side of the connection after passing through analog filters in the voice CODEC. |
| **digital** | Initiates the same loopback before passing through analog filters in the voice CODEC. |

## Default Values

No default value is necessary for this command.

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |

## Usage Examples

The following example activates a loopback toward the T1 network side of the connection after passing through analog filters in the voice CODEC:

(config)**#interface fxo 0/1**
(config-fxo 0/1)**#loopback analog**

# rx-gain <*value*>

Use the **rx-gain** command to define the receive gain characteristics on the FXO interface. Receive gain determines the amplification of the received signal before transmitting it out the FXO interface. Use the **no** form of this command to return to the default.

## Syntax Description

| | |
|---|---|
| <*value*> | Defines the receive gain characteristics for the interface in 0.1 decibel increments. Range is -6.0 to 10.0 dB. |

## Default Values

By default, this command is set to 0 dB.

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |

## Functional Notes

When increasing this value, the signal being received on this port sounds louder. When decreasing this value, the signal being received on this port sounds softer.

## Usage Examples

The following example defines the receive gain as -5.4 dB:

(config)**#interface fxo 0/1**
(config-fxo 0/1)**#rx-gain -5.4**

# test loop

Use the **test loop** command to manually control the FXO interface's hook switch. This is helpful when troubleshooting problems with the FXO equipment. Use the **no** form of this command to disable this feature. Variations of this command include:

**test loop closed**
**test loop open**

## Syntax Description

| | |
|---|---|
| **closed** | Closes the hook switch, allowing DC current to flow through the interface. |
| **open** | Opens the hook switch, preventing DC current from flowing through the interface. |

## Default Value

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |

## Usage Examples

The following example opens the interface's hook switch:

(config)**#interface fxo 02/1**
(config-fxo 0/1)**#test loop open**

# test ring-ground

Use the **test ring-ground** command to force the ring conductor to ground potential and provides battery on tip for detection of tip ground. This is helpful when troubleshooting problems with ground start circuits. Use the **no** form of this command to return to the default.

## Syntax Description

No subcommands.

## Default Value

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |

## Usage Examples

The following example forces a ring-ground test of the FXO interface:

(config)**#interface fxo 0/1**
(config-fxo 0/1)**#test ring-ground**

# test tip-ground

Use the **test tip-ground** command to detect the removal of the ring ground and check for the loop condition on an active FXO interface. This is helpful when troubleshooting problems with ground start circuits. Use the **no** form of this command to return to the default.

## Syntax Description

No subcommands.

## Default Value

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |

## Usage Examples

The following example forces a tip-ground test of the FXO interface:

```
(config)#interface fxo 0/1
(config-fxo 0/1)#test tip-ground
```

# test tone

Use the **test tone** command to activate the 1 kHz test tone. Use the **no** form of this command to deactivate the test tone. Variations of this command include:

**test tone far**
**test tone near**

## Syntax Description

| | |
|---|---|
| **far** | Sends the test tone out the T1 network interface to the remote end. |
| **near** | Sends the test tone toward the FXO interface. |

## Default Value

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |

## Usage Examples

The following example sends the test tone toward the FXO interface:

(config)**#interface fxo 0/1**
(config-fxo 0/1)**#test tone near**

# tx-gain <*value*>

Use the **tx-gain** command to define the transmit gain characteristics on the FXO interface. Transmit gain determines the amplification of the transmitted signal before transmitting from the FXO interface toward the network. Use the **no** form of this command to return to the default.

## Syntax Description

| | |
|---|---|
| <*value*> | Defines the transmit gain characteristics in 0.1 decibel increments. Range is -6.0 to 10.0 dB. |

## Default Value

By default, transmit gain is set to 0 dB.

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |

## Functional Notes

When increasing this value, the signal being transmitted to the far end sounds louder. When decreasing this value, the signal being transmitted to the far end sounds softer.

## Usage Examples

The following example defines the transmit gain as -5.4 dB on the FXO interface:

(config)**#interface fxo 0/1**
(config-fxo 0/1)**#tx-gain -5.4**

# FXS INTERFACE CONFIGURATION COMMAND SET

To activate the FXS Interface Configuration mode, enter the **interface fxs** command and specify the FXS port at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#interface fxs 2/1
(config-fxs 2/1)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:



All other commands for this command set are described in this section in alphabetical order.



# impedance

Use the **impedance** command to configure the AC impedance of the 2-wire interface. Use the **no** form of this command to return to the default value. Variations of this command include:

**impedance 600c**
**impedance 600r**
**impedance 900c**
**impedance 900r**

## Syntax Description

| | |
|---|---|
| **600c** | Specifies an impedance of 600 Ω **+** 2.16 μF**.** |
| **600r** | Specifies an impedance of 600 Ω real. |
| **900c** | Specifies an impedance of 900 Ω + 2.16 μF. |
| **900r** | Specifies an impedance of 900 Ω real. |

## Default Value

The default for this command is 600r.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Usage Examples

The following example sets the impedance to 600 Ω + 2.16 μF:

(config)**#interface fxs 2/1**
(config-fxs 2/1)**#impedance 600c**

# loopback

Use the **loopback** command to activate a loopback toward the T1 network side on the FXS module. Use the **no** form of this command to deactivate the loopback. Variations of this command include:

**loopback analog**
**loopback digital**

## Syntax Description

| | |
|---|---|
| **analog** | Initiates a loopback toward the T1 network side of the connection after passing through analog filters in the voice CODEC. |
| **digital** | Initiates the same loopback before passing through analog filters in the voice CODEC. |

## Default Values

No default value is necessary for this command.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Usage Examples

The following example activates a loopback toward the T1 network side of the connection after passing through analog filters in the voice CODEC:

(config)**#interface fxs 2/1**
(config-fxs 2/1)**#loopback analog**

# ring-voltage *<value>*

The **ring-voltage** command set the ring voltage for the FXS interface. Increasing the ring voltage, sends a stronger ring signal to the phones connected to this interface. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| *<value>* | Specifies a ring voltage. Select from 50, 60 or 70 Vrms. |

## Default Value

By default, ring-voltage is set to 50 Vrms.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Usage Example

The following example sets the ring-voltage to 60 Vrms:

(config)**#interface fxs 2/1**
(config-fxs 2/1)**#sring-voltage 60**

# rx-gain <*value*>

Use the **rx-gain** command to define the receive gain characteristics on the FXS interface. Receive gain determines the amplification of the received signal before transmitting out the FXS interface. Use the **no** form of this command to return to the default.

## Syntax Description

| | |
|---|---|
| <*value*> | Defines the receive gain characteristics for the interface in 0.1 decibel increments. Range is -12.0 to 6.0 dB. |

## Default Values

By default, this command is set to -3.0 dB.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Functional Notes

When increasing this value, the signal being received on this port sounds louder. When decreasing this value, the signal being received on this port sounds softer.

## Usage Examples

The following example defines the receive gain as -6.4 dB:

(config)**#interface fxs 2/1**
(config-fxs 2/1)**#rx-gain -6.4**

# signal

The **signal** command configures the signaling mode for the FXS interface. Use the **no** form of this command to return to the default. Variations of this command include:

**signal ground-start**
**signal loop-start**

## Syntax Description

| | |
|---|---|
| **ground-start** | Applies resistance to the tip conductor of the circuit to indicate an off-hook condition. |
| **loop-start** | Bridges the tip and ring to indicate an off-hook (seizing the line) condition. |

## Default Value

By default, this command is set to **loop-start**.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Functional Notes

This signaling mode must match the configuration of the network.

## Usage Example

The following example sets the signaling mode to loop-start:

(config)**#interface fxs 2/1**
(config-fxs 2/1)**#signal loop-start**

# test battery

Use the **test battery** command to provide battery on the 2-wire FXS interface. This is helpful when troubleshooting wiring problems with the FXS equipment. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Value

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Usage Examples

The following example provides battery on the 2-wire FXS interface:

(config)**#interface fxs 2/1**
(config-fxs 2/1)**#test battery**

# test reverse-battery

Use the **test reverse-battery** command to provide reverse battery polarity on the FXS interface. This is helpful when troubleshooting wiring problems with the FXS equipment. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Value

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Usage Examples

The following example provides reverse battery polarity on the FXS interface:

```
(config)#interface fxs 2/1
(config-fxs 2/1)#test reverse-battery
```

# test ringing

Use the **test ringing** command to activate ringing voltage on the 2-wire FXS interface (using a 2-seconds-on/4-seconds-off cadence). The **no** version of this command removes the ringing voltage from the interface.

## Syntax Description

No subcommands.

## Default Value

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Usage Examples

The following example activates ringing voltage on the 2-wire FXS interface:

(config)**#interface fxs 2/1**
(config-fxs 2/1)**#test ringing**

# test tip-open

Use the **test tip-open** command to provide battery on ring and a high impedance on tip. This is helpful when troubleshooting problems with ground-start interfaces. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Value

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Usage Examples

The following example provides battery on ring and a high impedance on tip on the FXS interface:

(config)**#interface fxs 2/1**
(config-fxs 2/1)**#test tip-open**

# test tone

Use the **test tone** command to activate the 1 kHz test tone. Use the **no** form of this command to deactivate the test tone. Variations of this command include:

**test tone near**
**test tone far**

## Syntax Description

| | |
|---|---|
| **near** | Sends the test tone toward the FXS interface. |
| **far** | Sends the test tone out the T1 network interface to the remote end. |

## Default Value

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Usage Examples

The following example sends the test tone toward the FXS interface:

(config)**#interface fxs 2/1**
(config-fxs 2/1)**#test tone near**

# tx-gain <*value*>

Use the **tx-gain** command to define the transmit gain characteristics (configured in 0.1 dB increments) on the FXS interface. Transmit gain determines the amplification of the received signal before transmitting from the FXS interface toward the network. Use the **no** form of this command to return to the default.

## Syntax Description

| | |
|---|---|
| <*value*> | Defines the transmit gain characteristics for the interface in 0.1 decibel increments. Range is -12.0 to 6.0 dB. |

## Default Value

By default, this command is set to -6.0 dB.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Functional Notes

When increasing this value, the signal being transmitted to the far end will sound louder. When decreasing this value, the signal being transmitted to the far end sounds softer.

## Usage Examples

The following example defines the transmit gain as -6.4 dB on the FXS interface:

(config)**#interface fxs 2/1**
(config-fxs 2/1)**#tx-gain -6.4**

# G.703 INTERFACE CONFIGURATION COMMAND SET

To activate the G.703 Interface Configuration mode, enter the **interface e1** command (and specify the G.703 port) at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#interface e1 1/2
(config-e1 1/2)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:



All other commands for this command set are described in this section in alphabetical order.



# coding

Use the **coding** command to configure the line coding for the G.703 physical interface. This setting must match the line coding supplied on the circuit by the PBX. Use the **no** form of this command to return to the default setting. Variations of this command include:

**coding ami**
**coding hdb3**

## Syntax Description

| | |
|---|---|
| **ami** | Configures the line coding for alternate mark inversion (AMI). |
| **hdb3** | Configures the line coding for high-density bipolar 3 (HDB3). |

## Default Values

By default, all E1 interfaces are configured with HDB3 line coding.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

The line coding configured in the unit must match the line coding of the E1 circuit. A mismatch will result in line errors (e.g., BPVs).

## Usage Examples

The following example configures the G.703 interface for AMI line coding:

(config)**#interface e1 1/2**
(config-e1 1/2)**#coding ami**

# framing crc4

Use the **framing** command to configure the framing format for the G.703 interface. This parameter should match the framing format set on the external device. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| **crc4** | Enables CRC4 bits to be transmitted in the outgoing data stream. Also, the received signal is checked for CRC4 errors. |

## Default Values

By default, CRC4 is enabled.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

The framing value must match the configuration of the E1 circuit. A mismatch will result in a loss of frame alarm.

## Usage Examples

The following example configures the G.703 interface for CRC4 framing:

(config)**#interface e1 1/2**
(config-e1 1/2)**#framing crc4**

# loopback network

Use the **loopback network** command to initiate a loopback on the interface toward the network. Use the **no** form of this command to deactivate the loopback. Variations of this command include:

**loopback network line**
**loopback network payload**

## Syntax Description

| | |
|---|---|
| **line** | Initiates a metallic loopback of the physical E1 network interface. |
| **payload** | Initiates a loopback of the E1 framer (CSU portion) of the E1 network interface. |

## Default Values

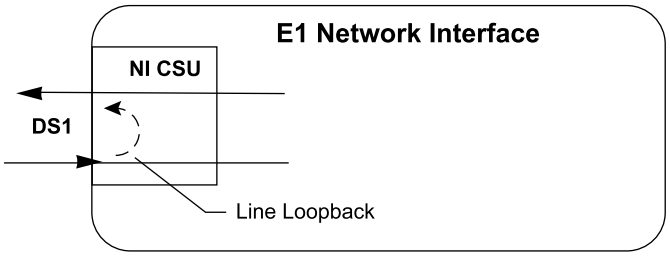
No default necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

The following diagram depicts a line loopback.




## Usage Examples

The following example initiates a line loopback of the G.703 interface:

(config)**#interface e1 1/2**
(config-e1 1/2)**#loopback network line**

# snmp trap link-status

Use the **snmp trap link-status** command to control the Simple Network Management Protocol (SNMP) variable ifLinkUpDownTrapEnable (RFC2863) to enable (or disable) the interface to send SNMP traps when there is an interface status change. Use the **no** form of this command to disable this trap.

## Syntax Description

No subcommands.

## Default Values

By default, the ifLinkUpDownTrapEnable OID is enabled for all interfaces except virtual Frame Relay interfaces.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 3.1 | Command was extended to the SHDSL interface. |
| Release 5.1 | Command was expanded to include Gigabit Ethernet, port channel, VLAN, E1, and G.703 interfaces. |

## Functional Notes

The **snmp trap link-status** command is used to control the RFC 2863 ifLinkUpDownTrapEnable OID (OID number 1.3.6.1.2.1.31.1.1.1.14.0).

## Usage Examples

The following example disables the link-status trap on the G.703 interface:

(config)**#interface e1 1/2**
(config-e1 1/2)**#no snmp trap link-status**

# test-pattern

Use the **test-pattern** command to activate the built-in pattern generator and begin sending the specified test pattern. This pattern generation can be used to verify a data path when used in conjunction with an active loopback. Use the **no** form of this command to cease pattern generation. Variations of this command include:

**test-pattern ones**
**test-pattern zeros**

## Syntax Description

| | |
|---|---|
| **ones** | Generates a test pattern of continous ones. |
| **zeros** | Generates a test pattern of continous zeros. |

## Default Values

No defaults necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |
| Release 6.1 | Command was expanded to include E1 and G.703 interfaces. |

## Usage Examples

The following example activates the pattern generator for a stream of continuous ones:

(config)**#interface e1 1/2**
(config-e1 1/2)**#test-pattern ones**

# ts16

Use the **ts16** command to enable timeslot 16 multiframe to be checked on the receive signal. Use the **no** form of this command to disable timeslot 16.

## Syntax Description

No subcommands.

## Default Values

No defaults necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following example enables timeslot 16 multiframing:

```
(config)#interface e1 1/2
(config-e1 1/2)#ts16
```

# HSSI INTERFACE CONFIGURATION COMMAND SET

To activate the HSSI Interface Configuration mode, enter the **interface hssi** command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#interface hssi 1/1
(config-hssi 1/1)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:



All other commands for this command set are described in this section in alphabetical order.



# external-loopback-request

Use the **external-loopback-request** command to enable LC (loopback circuit C) input to control loopbacks toward the network. Use the **no** form of this command to disable.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |

## Usage Examples

The following example enables the unit to accept external loopback requests:

(config)**#interface hssi 1/1**
(config-hssi 1/1)**#external-loopback-request**

# loopback

Use the **loopback** command to initiate or remove a loopback. Use the **no loopback** command to disable all loopbacks. Variations of this command include:

**loopback dce**
**loopback dte**
**loopback line**
**loopback remote**
**loopback none**

## Syntax Description

| | |
|---|---|
| **dce** | Initiates a loopback on the DCE. |
| **dte** | Initiates a loopback on the DTE. |
| **line** | Initiates a local line loopback. |
| **remote** | Initiates a remote line loopback. |
| **none** | Removes an active loopback. |

## Default Values

By default, no loopbacks are active.

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |

## Usage Examples

The following example initiates a local line loopback on the HSSI interface:

(config)**#interface hssi 1/1**
(config-hssi 1/1)**#loopback line**

# snmp trap link-status

Use the **snmp trap link-status** command to control the Simple Network Management Protocol (SNMP) variable ifLinkUpDownTrapEnable (RFC2863) to enable (or disable) the interface to send SNMP traps when there is an interface status change. Use the **no** form of this command to disable this trap.

## Syntax Description

No subcommands.

## Default Values

By default, the ifLinkUpDownTrapEnable OID is enabled for all interfaces except virtual Frame Relay interfaces.

## Functional Notes

The **snmp trap link-status** command is used to control the RFC2863 ifLinkUpDownTrapEnable OID (OID number 1.3.6.1.2.1.31.1.1.1.14.0).

## Usage Examples

The following example disables the link-status trap on the interface:

(config)**#interface hssi 1/1**
(config-hssi 1/1)**#no snmp trap link-status**

# MODEM INTERFACE CONFIGURATION COMMAND SET

To activate the Modem Interface Configuration mode, enter the **interface modem** command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#interface modem 1/2
(config-modem 1/2)#
```

*The modem interface number in the example above is shown as **modem 1/2**. This number is based on the interface's location (slot/port) and could vary depending on the unit's configuration. Use the **do show interfaces** command to determine the appropriate interface number.*

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:

*alias <"text"> on page 28*
*cross-connect on page 29*
*description <text> on page 32*
*do on page 33*
*end on page 34*
*exit on page 35*
*shutdown on page 36*

All other commands for this command set are described in this section in alphabetical order.

*caller-id override on page 951*
*dialin on page 952*
*init-string <string> on page 953*
*resource pool-member on page 954*

# caller-id override

Use the **caller-id override** command to configure the unit to replace caller ID information with a user-specified number. Use the **no** form of this command to disable any caller ID overrides. Variations of this command include:

**caller-id override always** <*number*>
**caller-id override if-no-cid** <*number*>

## Syntax Description

| | |
|---|---|
| **always** <*number*> | Always forces replacement of the incoming caller ID number with the number given. |
| **if-no-cid** <*number*> | Replaces the incoming caller ID number with the number given only if there is no caller ID information available for the incoming call. |

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Functional Notes

This command forces a replacement of the incoming caller ID number with the number given. The received caller ID, if any, is discarded, and the given override number is used to connect the incoming call to a circuit of the same number.

## Usage Examples

The following example configures the unit to always provide the given number as the caller ID number:

(config)**#interface modem 1/2**
(config-modem 1/2)**#caller-id override always 5555555**

# dialin

Use the **dialin** command to enable the modem for remote console dial-in, disabling the use of the modem for dial-backup. Use the **no** form to this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, **dialin** is disabled.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Usage Examples

The following example enables remote console dial-in:

(config)**#interface modem 1/2**
(config-modem 1/2)**#dialin**

# init-string *<string>*

Use the **init-string** command to specify an initialization string for the modem using standard AT commands. Use the **no** form of this command to return to the default initialization string.

## Syntax Description

| | |
|---|---|
| *<string>* | Specifies an initialization string using standard AT commands. This string must start with AT and cannot contain spaces. |

## Default Values

| | | |
|---|---|---|
| *<string>* | **ate0q0v1x4\n0** | |
| | **at** | All initialization strings must begin with AT. |
| | **e0** | Disables command echo. |
| | **q0** | Response messages on. |
| | **v1** | Formats result codes in long word form. |
| | **x4** | Specifies extended response set, dial tone, and busy signal detection for result codes following modem operations. |
| | **\n0** | Selects standard buffered connection only. |

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Usage Examples

The following example configures the modem to perform a hang-up at each initialization (to verify that the line is free) and maintains the default initialization:

(config)**#interface modem 1/2**
(config-modem 1/2)**#init-string ate0h0q0v1x4\n0**

# resource pool-member

Use the **resource pool-member** command to assign the interface to a resource pool, making it a demand routing resource. Use the **no** form of this command to return to the default value. Variations of this command include:

**resource pool-member** <*name*>
**resource pool-member** <*name*> <*cost*>

## Syntax Description

| | |
|---|---|
| <*name*> | Specifies the name of the resource pool to which this interface is assigned. |
| <*cost*> | Optional. Specifies the cost of using this resource interface within the specified pool. In the event of a tie, a resource with a lower cost will be selected first. Interfaces with the same cost will be selected in alphabetical order by interface name. |

## Default Values

By default, the interface is not assigned to any resource pool.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example configures a BRI interface as a member of resource pool **MyPool**:

(config)**#interface modem 1/2**
(config-modem 1/2)**#resource pool-member MyPool**

# PRI INTERFACE CONFIGURATION COMMAND SET

To activate the PRI Interface Configuration mode, enter the **interface pri** command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#interface pri 2
(config-pri 2)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:



All other commands for this command set are described in this section in alphabetical order.



# calling-party

Use the **calling party** command to configure and control the PRI outgoing caller ID information. Use the **no** form of these commands to disable these features. Variations of this command include:

**calling-party name** <*name*>
**calling-party number** <*number*>
**calling-party override always**
**calling-party override if-no-CID**
**calling-party override none**
**calling-party presentation allowed**
**calling-party presentation not-available**
**calling-party presentation restricted**

## Syntax Description

| | |
|---|---|
| **name** <*name*> | Configure the calling party name for the PRI. |
| **number** <*number*> | Configure the calling party number for the PRI. |
| **override always** | Enables the calling party to be replaced with the override number. |
| **override if-no-CID** | Enables the calling party to be replaced if caller ID no number is received. |
| **override none** | Sets the calling party override to none. |
| **presentation allowed** | Enables the presentation of caller ID to always be allowed. |
| **presentation not-available** | Sets the calling party number to not available. |
| **presentation restricted** | Restricts the delivery on the caller ID information. |

## Default Values

By default, the command is disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example configures calling party outgoing information:

(config)**#interface pri 2**
(config-pri 2)**#calling-party override always**
(config-pri 2)**#calling-party presentation 555-8000**
(config-pri 2)**#calling-party name Company, Inc.**

# connect t1 <*slot/port*> tdm-group <*number*>

Use the **connect t1** command to configure the TDM group connection used for the PRI interface. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*slot/port*> | Configure the T1 interface identifier. |
| <*number*> | Configure the TDM group number. Valid range is 1 to 255. |

## Default Values

By default, the command is disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example configures the unit to connect tdm-group 1 of the T1 to the PRI:

(config)**#interface pri 2**
(config-pri 2)**#connect t1 1/1 tdm-group 1**

# isdn name-delivery

Use the **isdn name-delivery** command to control the delivery of the name associated with the PRI. This command can be used to block the caller ID name on the PRI. Use the **no** form of this command to return to the default setting. Variations of this command include:

**isdn name-delivery none**
**isdn name-delivery proceeding**
**isdn name-delivery setup**

## Syntax Description

| | |
|---|---|
| **none** | Restricts the delivery of the calling party's name. |
| **proceeding** | Delivers the calling party's name in the proceeding message. |
| **setup** | Delivers the calling party's name in the setup message. |

## Default Values

By default, **isdn name-delivery** is set to **none**.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example configures the calling party information to be delivered in the setup message:

(config)**#interface pri 2**
(config-pri 2)**#isdn name-delivery setup**

# isdn switch-type

Use the **isdn switch-type** command to configure the switch type assigned on for PRI circuit. Telephone companies use various types of ISDN switches and this setting must match the switch type used by your provider. Use the **no** form of this command to return to the default setting. Variations of this command include:

**isdn switch-type 4ess**
**isdn switch-type 5ess**
**isdn switch-type dms100**
**isdn switch-type ni2**

## Syntax Description

| | |
|---|---|
| **4ess** | Sets the ISDN switch type to ATT 4ESS. |
| **5ess** | Sets the ISDN switch type to Lucent 5ESS. |
| **dms100** | Sets the ISDN switch type to Northern ISDN II. |
| **ni2** | Sets the ISDN switch type to National ISDN II. |

## Default Values

By default, the command is set to **ni2**.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example configures the PRI switch type National ISDN II:

(config)**#interface pri 2**
(config)**#isdn switch-type ni2**

# role

Use the **role** command to configure the interface protocol to use on the PRI. This setting controls the functional mode of the interface. Use the **no** form of this command to return to the default setting. Variations of this command include:

**role network**
**role network b-channel-restarts disable**
**role network b-channel-restarts enable**
**role user**

## Syntax Description

| | |
|---|---|
| **network** | Sets the port to operate in Network Termination (NT) mode. |
| **b-channel-restarts disable** | Optional. Disables B-channel restarts. |
| **b-channel-restarts enable** | Optional. Enables B-channel restarts. |
| **user** | Sets the port to operate in Terminal Equipment (TE) mode. |

## Default Values

By default, the role is set to **network b-channel-restarts disable**.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example configures the interface protocol as **user** on the PRI:

(config)**#interface pri 2**
(config)**#isdn user**

# SERIAL INTERFACE CONFIGURATION COMMAND SET

To activate the Serial Interface Configuration mode, enter the **interface serial** command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#interface serial 1/1
(config-ser 1/1)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:



All other commands for this command set are described in this section in alphabetical order.



# et-clock-source

Use the **et-clock-source** command to configure the clock source used when creating the external transmit reference clock (et-clock). Use the **no** form of this command to return to the default value. Variations of this command include:

**et-clock-source rxclock**
**et-clock-source txclock**

## Syntax Description

| | |
|---|---|
| **rxclock** | Uses the clock recovered from the receive signal to generate et-clock. |
| **txclock** | Uses the clock recovered from the transmit signal to generate et-clock. |

## Default Values

By default, the clock recovered from the transmit signal is used to generate the et-clock.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

The external transmit clock (et-clock) is an interface timing signal (provided by the DTE device) used to synchronize the transfer of transmit data.

## Usage Examples

The following example configures the serial interface to recover the clock signal from the received signal and use it to generate et-clock:

(config)**#interface serial 1/1**
(config-ser 1/1)**#et-clock-source rxclock**

# ignore dcd

Use the **ignore dcd** command to specify the behavior of the serial interface when the Data Carrier Detect (DCD) signal is lost. Use the **no** form of this command to return to the default value.

## Syntax Description

No subcommands.

## Default Values

By default, the serial interface does not ignore a change in status of the DCD signal.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

When configured to follow DCD (default condition), the serial interface will not attempt to establish a connection when DCD is not present. When configured to ignore DCD, the serial interface will continue to attempt to establish a connection even when DCD is not present.

## Usage Examples

The following example configures the serial interface to ignore a loss of the DCD signal:

(config)**#interface serial 1/1**
(config-ser 1/1)**#ignore dcd**

# invert etclock

Use the **invert etclock** command to configure the serial interface to invert the external transmit reference clock (et-clock) in the data stream before transmitting. Use the **no** form of this command to return to the default value.

## Syntax Description

No subcommands.

## Default Values

By default, the serial interface does not invert et-clock.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

If the serial interface cable is long, causing a phase shift in the data, the et-clock can be inverted using the **invert etclock** command. This switches the phase of the clock, which compensates for a long cable.

## Usage Examples

The following example configures the serial interface to invert et-clock:

(config)**#interface serial 1/1**
(config-ser 1/1)**#invert etclock**

# invert rxclock

Use the **invert rxclock** command to configure the serial interface to expect an inverted receive clock (found in the received data stream). Use the **no** form of this command to return to the default value.

## Syntax Description

No subcommands.

## Default Values

By default, the serial interface does not expect an inverted receive clock (**rxclock**).

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

If the serial interface cable is long, causing a phase shift in the data, the transmit clock can be inverted using the **invert txclock** command (refer to *invert txclock* on page 966). This switches the phase of the clock, which compensates for a long cable. If the transmit clock of the connected device is inverted, use the **invert rxclock** command to configure the receiving interface appropriately.

## Usage Examples

The following example configures the serial interface to invert receive clock:

(config)**#interface serial 1/1**
(config-ser 1/1)**#invert rxclock**

# invert txclock

Use the **invert txclock** command to configure the serial interface to invert the transmit clock (found in the transmitted data stream) before sending the signal. Use the **no** form of this command to return to the default value.

## Syntax Description

No subcommands.

## Default Values

By default, the serial interface does not invert transmit clock (**txclock)**.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

If the serial interface cable is long, causing a phase shift in the data, the transmit clock can be inverted (using the **invert txclock** command). This switches the phase of the clock, which compensates for a long cable. If the transmit clock of the connected device is inverted, use the **invert rxclock** command to configure the receiving interface appropriately.

## Usage Examples

The following example configures the serial interface to invert the transmit clock:

(config)**#interface serial 1/1**
(config-ser 1/1)**#invert txclock**

# serial-mode

Use the **serial-mode** command to specify the electrical mode for the interface. Use the **no** form of this command to return to the default value. Variations of this command include:

**serial-model eia530**
**serial-model v35**
**serial-model x21**

## Syntax Description

| | |
|---|---|
| **eia530** | Configures the interface for use with the EIA 530 adapter cable (P/N 1200883L1). |
| **v35** | Configures the interface for use with the V.35 adapter cable (P/N 1200873L1). |
| **x21** | Configures the interface for use with the X.21 adapter cable (P/N 1200874L1). |

## Default Values

By default, the serial interface is configured for a V.35 adapter cable.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

The pinouts for each of the available interfaces can be found in the *Hardware Configuration Guide* located on the *ADTRAN OS System Documentation* CD (provided in shipment).

## Usage Examples

The following example configures the serial interface to work with the X.21 adapter cable:

(config)**#interface serial 1/1**
(config-ser 1/1)**#serial-mode X21**

# snmp trap

Use the **snmp trap** command to enable all supported Simple Network Management Protocol (SNMP) traps on the interface. Use the **no** form of this command to disable this trap.

## Syntax Description

No subcommands.

## Default Values

By default, all interfaces (except virtual Frame Relay interfaces and sub-interfaces) have SNMP traps enabled.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 3.1 | Command was extended to the SHDSL interface. |
| Release 5.1 | Command was expanded to include Ethernet sub-interfaces and Gigabit Ethernet interfaces. |

## Usage Examples

The following example enables SNMP on the serial interface:

(config)**#interface serial 1/1**
(config-ser 1/1)**#snmp trap**

# snmp trap link-status

Use the **snmp trap link-status** command to control the Simple Network Management Protocol (SNMP) variable ifLinkUpDownTrapEnable (RFC2863) to enable (or disable) the interface to send SNMP traps when there is an interface status change. Use the **no** form of this command to disable this trap.

## Syntax Description

No subcommands.

## Default Values

By default, the ifLinkUpDownTrapEnable OID is enabled for all interfaces except virtual Frame Relay interfaces.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 3.1 | Command was extended to the SHDSL interface. |
| Release 5.1 | Command was expanded to include Gigabit Ethernet, port channel, VLAN, E1, and G.703 interfaces. |

## Functional Notes

The **snmp trap link-status** command is used to control the RFC 2863 ifLinkUpDownTrapEnable OID (OID number 1.3.6.1.2.1.31.1.1.1.14.0).

## Usage Examples

The following example disables the link-status trap on the serial interface:

(config)**#interface serial 1/1**
(config-ser 1/1)**#no snmp trap link-status**

# SHDSL INTERFACE CONFIGURATION COMMAND SET

To activate the SHDSL Interface Configuration mode, enter the **interface shdsl** command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config#)interface shdsl 1/1
(config-shdsl 1/1)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:



All other commands for this command set are described in this section in alphabetical order.



# alarm-threshold

Use the **alarm-threshold** command to set thresholds for specific alarm conditions. Use the **no** form of this command to disable threshold settings. Variations of this command include:

**alarm-threshold loop-attenuation** <*value*>
**alarm-threshold snr-margin** <*value*>

## Syntax Description

| | |
|---|---|
| **loop-attenuation** <*value*> | Specifies a loop-attenuation threshold value from 1 to 127 dB. If signal energy loss on the loop exceeds the configured value, the router issues an alarm. |
| **snr-margin** <*value*> | Specifies a value for signal-to-noise ratio (SNR) margin from 1 to 15 dB. If the difference in amplitude between the baseband signal and the noise exceeds the configured value, the router issues an alarm. |

## Default Values

No defaults necessary for this command.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Usage Examples

The following example sets the loop attenuation threshold at 45 dB:

(config)**#interface shdsl 1/1**
(config-shdsl 1/1)**#alarm-threshold loop-attenuation 45**

# boot alternate-image

Use the **boot alternate-image** command to execute new code after a firmware upgrade.

## Syntax Description

No subcommands.

## Default Values

No defaults necessary for this command.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Functional Notes

The current SHDSL NIM card (P/N 1200867L1) supports two code images commonly referred to as the “active” image and the “inactive” image. When a firmware upgrade is performed on the card (through the **copy <filename> interface shdsl x/y** Enable mode command), the new firmware is placed in the “inactive” image space. This new code will not be executed until the **boot alternate-image** command is issued. When the user does this, the NIM will reboot (taking the current line down) with the new code. At this point, the old code becomes the “inactive” image and the new recently updated code becomes the “active” image.

## Usage Examples

The following example causes the firmware upgrade to take effect:

(config)**#interface shdsl 1/1**
(config-shdsl 1/1)**#boot alternate-image**

# equipment-type

Use the **equipment-type** command to determine the operating mode for the SHDSL interface. Use the **no** form of this command to return to the default setting. Variations of this command include:

**equipment-type co**
**equipment-type pe**

## Syntax Description

| | |
|---|---|
| **co** | Use this option only in a campus environment when operating two SHDSL network interface modules (NIMs) back-to-back. In this setup, configure the master NIM to CO and the slave NIM to CPE. |
| **cpe** | Use this option when interfacing directly with your service provider or when acting as the slave NIM in a campus environment. |

## Default Values

The default for this command is **cpe**.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Usage Examples

The following example changes the operating mode of the SHDSL interface to CO:

(config)**#interface shdsl 1/1**
(config-shdsl 1/1)**#equipment-type co**

# inband-detection

Use the **inband-detection** command to enable inband loopback pattern detection on the SHDSL interface. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, this command is enabled.

## Command History

| | |
|---|---|
| Release 4.1 | Command was introduced. |

## Usage Examples

The following example disables inband loopback pattern detection:

```
(config)#interface shdsl 1/1
(config-shdsl 1/1)#no inband-detection
```

# inband-protocol

Use the **inband-protocol** command to designate the inband loopback pattern to send/detect on the SHDSL interface. Use the **no** form of this command to return to default. Variations of this command include:

**inband-protocol pn127**
**inband-protocol v54**

## Syntax Description

| | |
|---|---|
| **pn127** | Selects PN127 as the inband loopback pattern to send/detect. |
| **v54** | Selects V.54 as the inband loopback pattern to send/detect. |

## Default Values

By default, the inband loopback pattern is set to **v54**.

## Command History

| | |
|---|---|
| Release 4.1 | Command was introduced. |

## Functional Notes

Inband loopbacks are specific patterns that are sent in place of user data to trigger a loopback. Both PN127 and V.54 are industry-standard loopback patterns used to allow remote loopbacks.

## Usage Examples

The following example sets the inband loopback pattern for PN127:

(config)**#interface shdsl 1/1**
(config-shdsl 1/1)**#inband-protocol pn127**

# linerate <*value*>

Use the **linerate** command to define the line rate for the SHDSL interface (the value includes 8 kbps of framing overhead). This command is functional only in CO operating mode (refer to *equipment-type* on page 973). The first two selections listed in the CLI (72 and 136 kbps) are not supported by the SHDSL NIM (P/N 1200867L1). Use the **no** form of this command to return to the default setting.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the line rate in kbps. Range is 200 to 2312 kbps in 64k increments. |

## Default Values

By default, the line rate is set to 2056 kbps.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Usage Examples

The following example changes the line rate of the SHDSL interface to 264 kbps:

(config)**#interface shdsl 1/1**
(config-shdsl 1/1)**#linerate 264**

# loopback network

Use the **loopback network** command to initiate a loopback test on the SHDSL interface, looping the data toward the network. Use the **no** form of this command to deactivate the loopback.

## Syntax Description

No subcommands.

## Default Values

No default necessary for this command.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Usage Examples

The following example initiates a loopback on the SHDSL interface:

```
(config)#interface shdsl 1/1
(config-shdsl 1/1)#loopback network
```

# loopback remote

Use the **loopback remote** command to send a loopback request to the remote unit. This command is functional only in CO operating mode (refer to *equipment-type* on page 973). Use the **no** form of this command to send a loopdown code to the remote unit to deactivate the loopback.

## Syntax Description

No subcommands.

## Default Values

No defaults necessary for this command.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Usage Examples

The following example initiates a remote line loopback:

(config)**#interface shdsl 1/1**
(config-shdsl 1/1)**#loopback remote**

# loopback remote inband

Use the **loopback remote inband** command to inject the selected inband loop-up pattern into the data stream to cause a loopback at the far end. Use the **no** form of this command to inject a loop-down pattern into the data stream to cause an existing inband loopback at the far end to cease.

## Syntax Description

No subcommands.

## Default Values

By default, this command is enabled.

## Command History

| | |
|---|---|
| Release 4.1 | Command was introduced. |

## Usage Examples

The following example injects a loop-down pattern into the data stream, causing existing loopbacks at the far end to stop:

(config)**#interface shdsl 1/1**
(config-shdsl 1/1)**#no loopback remote inband**

# outage-retrain

Use the **outage-retrain** command to cause the SHDSL interface to force the SHDSL retrain sequence (which takes the line down temporarily) if the interface detects more than ten consecutive errored seconds. A retrain is forced in hopes that the newly retrained line will achieve better performance than the previous training state. Use the **no** version of the command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, this feature is disabled.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Usage Examples

The following example forces a retrain sequence on the SHDSL interface:

```
(config)#interface shdsl 1/1
(config-shdsl 1/1)#outage-retrain
```

# test-pattern

Use the **test-pattern** command to activate the built-in pattern generator and begin sending the selected test pattern toward the network. This pattern generation can be used to verify a data path when used in conjunction with an active loopback. Use the **no** form of this command to cease pattern generation. Variations of this command include:

**test-pattern clear**
**test-pattern errors**
**test-pattern insert**
**test-pattern p215**

## Syntax Description

| | |
|---|---|
| **clear** | Clears the test pattern error count. |
| **errors** | Displays the test pattern error count. |
| **insert** | Inserts an error into the currently active test pattern. |
| **p215** | Generates a pseudorandom test pattern sequence based on a 15-bit shift register. |

## Default Values

No defaults necessary for this command.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Usage Examples

The following example sends a $2^{15}$ test pattern:

(config)**#interface shdsl 1/1**
(config-shdsl 1/1)**#test-pattern p215**

# T1 INTERFACE CONFIGURATION COMMAND SET

To activate the T1 Interface Configuration mode, enter the **interface t1** command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#interface t1 1/1
(config-t1 1/1)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:



All other commands for this command set are described in this section in alphabetical order.



# clock source

Use the **clock source** command to configure the source timing used for the interface. Use the **no** form of this command to return to the default value. Variations of this command include:

**clock source internal**
**clock source line**
**clock source system**
**clock source through**
**clock source through t1** *<interface id>*

## Syntax Description

| | |
|---|---|
| **internal** | Configures the unit to provide clocking using the internal oscillator. |
| **line** | Configures the unit to recover clocking from the T1 circuit. |
| **system** | Configures the unit to provide clocking using the system clock. |
| **through** | Configures the unit to recover clocking from the circuit connected to the DSX-1 interface. |
| **through t1** *<interface id>* | Configures the unit to recover clocking from the alternate interface. Only valid on T1 systems with multiple T1 interfaces and a single clock source. |

## Default Values

By default, the **clock source** is set to **line**.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 13.1 | Command was expanded to include system as a clocking source. |

## Functional Notes

When operating on a circuit that is providing timing, setting the **clock source** to **line** can avoid errors such as Clock Slip Seconds (CSS).

## Usage Examples

The following example configures the unit to recover clocking from the primary circuit:

(config)**#interface t1 1/1**
(config-t1 1/1)**#clock source line**

# coding

Use the **coding** command to configure the line coding for a T1 physical interface. This setting must match the line coding supplied on the circuit by the service provider. Use the **no** form of this command to return to the default setting. Variations of this command include:

**coding ami**
**coding b8zs**

## Syntax Description

| | |
|---|---|
| **ami** | Configures the line coding for alternate mark inversion (AMI). |
| **b8zs** | Configures the line coding for bipolar eight zero substitution (B8ZS). |

## Default Values

By default, all T1 interfaces are configured with B8ZS line coding.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

The line coding configured in the unit must match the line coding of the T1 circuit. A mismatch will result in line errors (e.g., BPVs).

## Usage Examples

The following example configures the T1 interface for AMI line coding:

(config)**#interface t1 1/1**
(config-t1 1/1)**#coding ami**

# fdl

Use the **fdl** command to configure the format for the facility data link (FDL) channel on the T1 circuit. FDL channels are only available on point-to-point circuits. Use the **no** form of this command to return to the default value. Variations of this command include:

**fdl ansi**
**fdl att**
**fdl none**

## Syntax Description

| | |
|---|---|
| **ansi** | Configures the FDL for ANSI T1.403 standard. |
| **att** | Configures the FDL for AT&T TR 54016 standard. |
| **none** | Disables FDL on this circuit. |

## Default Values

By default, the FDL is configured for **ansi**.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

T1 circuits using ESF framing format (specified using the **framing** command) reserve 12 bits as a data link communication channel, referred to as the FDL, between the equipment on either end of the circuit. The FDL allows the transmission of trouble flags such as the Yellow Alarm signal. Refer to *framing* on page 986 for related information.

## Usage Examples

The following example disables t**he FDL channel for the T1 circuit:**

**(config)#**interface t1 1/1
(config-t1 1/1)**#fdl none**

# framing

Use the **framing** command to configure the framing format for the T1 interface. This parameter should match the framing format supplied by your network provider. Use the **no** form of this command to return to the default value. Variations of this command include:

**framing d4**
**framing esf**

## Syntax Description

| | |
|---|---|
| **d4** | Specifies D4 superframe (SF) format. |
| **esf** | Specifies extended superframe (ESF) format. |

## Default Values

By default, the framing format is set to **esf**.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

A frame is comprised of a single byte from each of the T1's timeslots; there are 24 timeslots on a single T1 circuit. Framing bits are used to separate the frames and indicate the order of information arriving at the receiving equipment. D4 and ESF are two methods of collecting and organizing frames over the circuit.

## Usage Examples

The following example configures the T1 interface for D4 framing:

(config)**#interface t1 1/1**
(config-t1 1/1)**#framing d4**

# lbo

Use the **lbo** command to configure the line build out (LBO) for the T1 interface. Use the **no** form of this command to return to the default value. Variations of this command include:

**lbo short** <*value*>
**lbo long** <*value*>

## Syntax Description

| | |
|---|---|
| **long** <*value*> | Configures the LBO (in dB) for T1 interfaces with cable lengths greater than 655 feet. Choose from -22.5, -15, -7.5, and 0 dB. |
| **short** <*value*> | Configures the LBO (in feet) for T1 interfaces with cable lengths less than 655 feet. Range is 0 to 655 feet. |

## Default Values

By default, the build out is set to 0 dB.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

Line build out (LBO) is artificial attenuation of a T1 output signal to simulate a degraded signal. This is useful to avoid overdriving a receiver's circuits. The shorter the distance between T1 equipment (measured in cable length), the greater the attenuation value. For example, two units in close proximity should be configured for the maximum attenuation (-22.5 dB).

## Usage Examples

The following example configures the T1 interface LBO for -22.5 dB:

(config)**#interface t1 1/1**
(config-t1 1/1)**#lbo -22.5**

# loopback network

Use the **loopback network** command to initiate a loopback on the interface toward the network. Use the **no** form of this command to deactivate the loopback. Variations oft his command include:

**loopback network line**
**loopback network payload**

## Syntax Description

| | |
|---|---|
| **line** | Initiates a metallic loopback of the physical T1 network interface. |
| **payload** | Initiates a loopback of the T1 framer (CSU portion) of the T1 network interface. |

## Default Values

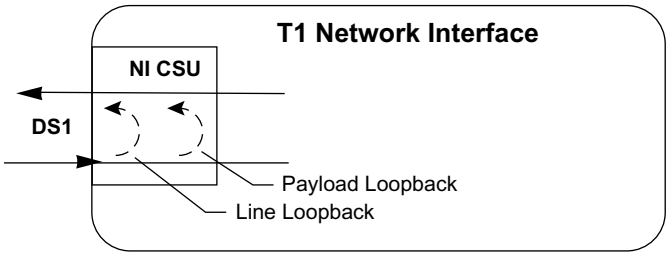
No default necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

The following diagram depicts the difference between a line and payload loopback.




## Usage Examples

The following example initiates a payload loopback of the T1 interface:

(config)**#interface t1 1/1**
**(config-t1 1/1)#**loopback network payload

# loopback remote line

Use the **loopback remote line** command to send a loopback code to the remote unit to initiate a line loopback. Use the **no** form of this command to send a loopdown code to the remote unit to deactivate the loopback. Variations of this command include:

**loopback remote line fdl**
**loopback remote line inband**

## Syntax Description

| | |
|---|---|
| **fdl** | Uses the facility data link (FDL) to initiate a full 1.544 Mbps physical (metallic) loopback of the signal received by the remote unit from the network. |
| **inband** | Uses the inband channel to initiate a full 1.544 Mbps physical (metallic) loopback of the signal received by the remote unit from the network. |

## Default Values

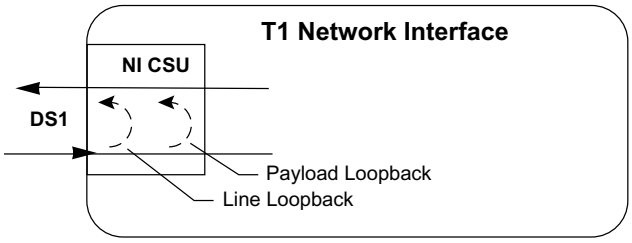
No defaults necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

The following diagram depicts the difference between a line and payload loopback.




## Usage Examples

The following example initiates a remote line loopback using the FDL:

(config)**#interface t1 1/1**
(config-t1 1/1)**#loopback remote line fdl**

# loopback remote payload

Use the **loopback remote payload** command to send a loopback code to the remote unit to initiate a payload loopback. A payload loopback is a 1.536 Mbps loopback of the payload data received from the network maintaining bit-sequence integrity for the information bits by synchronizing (regenerating) the timing. Use the **no** form of this command to send a loopdown code to the remote unit to deactivate the loopback.

## Syntax Description

No subcommands.

## Default Values

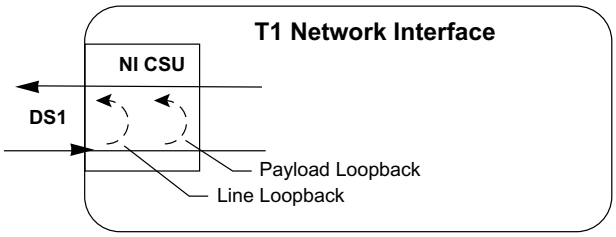
No defaults necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

The following diagram depicts the difference between a line and payload loopback.




## Usage Examples

The following example initiates a remote payload loopback:

(config)**#interface t1 1/1**
(config-t1 1/1)**#loopback remote payload**

# remote-alarm rai

The **remote-alarm rai** command selects the alarm signaling type to be sent when a loss of frame is detected on the T1 receive signal. Use the **no** form of this command to disable all transmitted alarms.

## Syntax Description

| | |
|---|---|
| **rai** | Specifies sending a remote alarm indication (RAI) in response to a loss of frame. Also prevents a received RAI from causing a change in interface operational status. |

## Default Values

The default for this command is **rai**.

## Command History

| | |
|---|---|
| Release 7.1 | Command was expanded to include the T1 interface. |

## Usage Examples

The following example enables transmission of RAI in response to a loss of frame:

(config)**#interface t1 1/1**
(config-t1 1/1)**#remote-alarm rai**

# remote-loopback

Use the **remote-loopback** command to configure the interface to respond to loopbacks initiated by a remote unit (or the service provider). Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, all interfaces respond to remote loopbacks.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example enables remote loopbacks on the T1 interface:

```
(config)#interface t1 1/1
(config-t1 1/1)#remote-loopback
```

# snmp trap line-status

Use the **snmp trap line-status** command to control the Simple Network Management Protocol (SNMP) variable dsx1LineStatusChangeTrapEnable (RFC2495) to enable (or disable) the interface to send SNMP traps when there is an interface status change. Use the **no** form of this command to disable this trap.

## Syntax Description

No subcommands.

## Default Values

By default, the dsx1LineStatusChangeTrapEnable OID is set to enabled for all interfaces except virtual Frame Relay Interfaces.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Functional Notes

The **snmp trap line-status** command is used to control the RFC2495 dsx1LineStatusChangeTrapEnable OID (OID number 1.3.6.1.2.1.10.18.6.1.17.0).

## Usage Examples

The following example disables the line-status trap on the T1 interface:

(config)**#interface t1 1/1**
(config-t1 1/1)**#no snmp trap line-status**

# snmp trap link-status

Use the **snmp trap link-status** command to control the Simple Network Management Protocol (SNMP) variable ifLinkUpDownTrapEnable (RFC2863) to enable (or disable) the interface to send SNMP traps when there is an interface status change. Use the **no** form of this command to disable this trap.

## Syntax Description

No subcommands.

## Default Values

By default, the ifLinkUpDownTrapEnable OID is enabled for all interfaces except virtual Frame Relay interfaces.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 3.1 | Command was extended to the SHDSL interface. |
| Release 5.1 | Command was expanded to include Gigabit Ethernet, port channel, VLAN, E1, and G.703 interfaces. |

## Functional Notes

The **snmp trap link-status** command is used to control the RFC2863 ifLinkUpDownTrapEnable OID (OID number 1.3.6.1.2.1.31.1.1.1.14.0).

## Usage Examples

The following example disables th**e link-status trap on the T1 interface:**

**(config)#**interface t1 1/1
**(config-t1 1/1)#**no snmp trap link-status

# snmp trap threshold-reached

Use the **snmp trap threshold-reached** command to control the Simple Network Management Protocol (SNMP) variable adGenAOSDs1ThresholdReached (adGenAOSDs1-Ext MIB) to enable the interface to send SNMP traps when a DS1 performance counter threshold is reached. Use the **no** form of this command to disable this trap.

## Syntax Description

No subcommands.

## Default Values

By default, the adGenAOSDs1ThresholdReached OID is enabled for all interfaces.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example disables SNMP threshold reached trap on the T1 interface:

(config)**#interface t1 1/1**
(config-t1 1/1)**#no snmp trap threshold-reached**

# system timing

Use the **system timing** command to configure the Rx clock as the primary or secondary timing source for the system. Use the **no** form of this command to disable this feature.

## Syntax Description

| | |
|---|---|
| **primary** | Specifies the Rx clock as the primary timing source. |
| **secondary** | Specifies the Rx clock as the secondary timing source. |

## Default Values

There is no default for this command.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example configures the T1 interface to provide its Rx clock as the primary timing source for the system:

(config)**#interface t1 1/1**
(config-t1 1/1)**#system timing primary**

# tdm-group *<group number>*

Use the **tdm-group** command to create a group of contiguous DS0s on this interface to be used during the **cross-connect** process. Refer to *cross-connect* on page 29 for related information. Use the **no** form of this command to remove configured TDM groups. Variations of this command include:

**tdm-group** <*number*> **timeslots** <*value*>
**tdm-group** <*number*> **timeslots** <*value*> **speed [56 | 64]**


*Changing **tdm-group** settings could result in service interruption.*

## Syntax Description

| | |
|---|---|
| <*number*> | Identifies the created TDM group. Valid range is 1 to 255. |
| **timeslots** <*value*> | Specifies the channels to be used in the TDM group. Valid range is 1 to 31. The timeslot value can be entered as a single number representing one of the 31 E1 channel timeslots or as a contiguous group of channels. (For example, **1-10** specifies the first 10 channels of the E1.) |
| **speed [56 \| 64]** | Optional. Specifies the individual DS0 rate on the T1 interface to be 64 kbps. Only the T1 + DSX-1 Network Interface Module supports the 56 kbps DS0 rate. The default speed is 64 kbps. |

## Default Values

By default, there are no configured TDM groups.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example creates a TDM group (labeled **5**) of 10 DS0s at 64 kbps each:

(config)**#interface t1 1/1**
(config-t1 1/1)**#tdm-group 5 timeslots 1-10 speed 64**

# test-pattern

Use the **test-pattern** command to activate the built-in pattern generator and begin sending the specified test pattern. This pattern generation can be used to verify a data path when used in conjunction with an active loopback. Use the **no** form of this command to cease pattern generation. Variations of this command include:

**test-pattern clear**
**test-pattern errors**
**test-pattern insert**
**test-pattern ones**
**test-pattern p215**
**test-pattern p220**
**test-pattern p511**
**test-pattern qrss**
**test-pattern zeros**

## Syntax Description

| | |
|---|---|
| **clear** | Clears the test pattern error count. |
| **errors** | Displays the test pattern errored seconds. |
| **insert** | Inserts an error into the currently active test pattern. |
| **ones** | Generates a test pattern of continous ones. |
| **p215** | Generates a pseudorandom test pattern sequence based on a 15-bit shift register. |
| **p220** | Generates a pseudorandom test pattern sequence based on a 20-bit shift register. |
| **p511** | Generates a test pattern of repeating ones and zeros. |
| **qrss** | Generates a test pattern of random ones and zeros. |
| **zeros** | Generates a test pattern of continous zeros. |

## Default Values

No defaults necessary for this command.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example activates the pattern generator for a stream of continuous ones:

(config)**#interface t1 1/1**
(config-t1 1/1)**#test-pattern ones**

# T3 INTERFACE CONFIGURATION COMMAND SET

To activate the T3 Interface Configuration mode, enter the **interface t3** command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#interface t3 1/1
(config-t3 1/1)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:



All other commands for this command set are described in this section in alphabetical order.



# clock source

Use the **clock source** command to configure the source timing used for the interface. The clock specified using the **clock source** command is also the system master clock. Use the **no** form of this command to return to the default value. Variations of this command include:

**clock source local**
**clock source loop**

## Syntax Description

| | |
|---|---|
| **local** | Configures the unit to provide clocking using the internal oscillator. |
| **loop** | Configures the unit to recover clocking from the T3 circuit. |

## Default Value

By default, all T3 interfaces are configured with loop as the clock source.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Usage Examples

The following example configures the unit to recover clocking from the circuit:

(config)**#interface t3 1/1**
(config-t3 1/1)**#clock source loop**

# coding b3zs

Use the **coding b3zs** command to configure the line coding for a T3 physical interface. This setting must match the line coding supplied on the circuit by the service provider.

## Syntax Description

| | |
|---|---|
| **b3zs** | Configures the line coding for bipolar three zero substitution (B3ZS). |

## Default Value

By default, all T3 interfaces are configured with B3ZS line coding.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Functional Notes

The line coding configured in the unit must match the line coding of the T3 circuit. A mismatch will result in line errors (e.g., BPVs).

## Usage Examples

The following example configures the T1 interface for B3ZS line coding:

(config)**#interface t3 1/1**
(config-t3 1/1)**#coding b3zs**

# framing

Use the **framing** command to configure the network framing format for a T3 physical interface. Use the **no** form of this command to return to the default value. Variations of this command include:

**framing cbit**
**framing m13**

## Syntax Description

| | |
|---|---|
| **cbit** | Configures the interface for C-bit parity framing. |
| **m13** | Configures the interface for M13 framing. |

## Default Value

By default, all T3 interfaces are configured for C-bit parity framing.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Functional Notes

M13 is an asynchronous framing format that uses all 21 DS3 M-Frame C-bits for bit stuffing indicators. End-to-end path parity and datalink capabilities are not provided by the standard M13 format. C-bit parity framing differs from M13 by allowing monitoring of the data path (end-to-end) and supporting out-of-band data links.

## Usage Examples

The following example configures the T3 interface for M13 framing:

(config)**#interface t3 1/1**
(config-t3 1/1)**#framing m13**

# line-length

Use the **line-length** command to configure the line length for a T3 physical interface. Use the no form of this command to return to the default value. Variations of this command include:

**line-length long**
**line-length short**

## Syntax Description

| | |
|---|---|
| **long** | Configures the line length for a distance of 225 to 450 feet of cable. |
| **short** | Configures the line length for a distance of 0 to 225 feet of cable. |

## Default Value

By default, all T3 interfaces are configured for a **short** line length.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Usage Examples

The following example configures the T3 interface for long line length:

(config)**#interface t3 1/1**
(config-t3 1/1)**#line-length long**

# loopback network

Use the **loopback network** command to initiate a local T3 loopback on the interface toward the network. Use the **no** form of this command to deactivate the loopback. Variations of this command include:

**loopback network line**
**loopback network payload**

## Syntax Description

| | |
|---|---|
| **line** | Initiates a loopback of the physical T3 network interface; that is, data received on the T3 is transmitted back out on the T3. |
| **payload** | Initiates a loopback of the T3 framer (TSU portion) of the T3 network interface. |

## Default Value

No default necessary for this command.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Usage Examples

The following example initiates a payload loopback of the T3 interface:

(config)**#interface t3 1/1**
(config-t3 1/1)**#loopback network payload**

# loopback remote

Use the **loopback remote** command to initiate a loopback test on the T3 interface that sends a remote loopback code out the T3 circuit to loop up the far end. This command only applies when C-bit framing is used on the circuit. Use the **no** form of this command to deactivate the loopback. Variations of this command include:

**loopback remote line**
**loopback remote payload**

## Syntax Description

| | |
|---|---|
| **line** | Initiates a line loopback. |
| **payload** | Initiates a payload loopback. |

## Default Value

No default value is necessary for this command.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Usage Examples

This example initiates a remote loopback on the T3 interface:

(config)**#interface t3 1/1**
(config-t3 1/1)**#loopback remote**

# remote-loopback

Use the **remote-loopback** command to configure the T3 interface to be looped *from* the far end (remote device, telco, etc.). Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Value

By default, all interfaces respond to remote loopbacks.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Usage Examples

This example enables remote loopbacks on the T3 interface:

```
(config)#interface t3 1/1
(config-t3 1/1)#remote-loopback
```

# snmp trap link-status

Use the **snmp trap link-status** command to control the Simple Network Management Protocol (SNMP) variable ifLinkUpDownTrapEnable (RFC2863) to enable (or disable) the interface to send SNMP traps when there is an interface status change. Use the **no** form of this command to disable this trap.

## Syntax Description

No subcommands.

## Default Value

By default, the ifLinkUpDownTrapEnable OID is enabled for all interfaces except virtual Frame Relay interfaces.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Usage Examples

The following example disables the link-status trap on the T3 interface:

(config)**#interface t3 1/1**
(config-t3 1/1)**#no snmp trap link-status**

# test-pattern

Use the **test-pattern** command to activate the built-in pattern generator and begin sending the selected test pattern toward the network. This pattern generation can verify a data path when used in conjunction with an active loopback. Use the **no** form of this command to cease pattern generation. Variations of this command include:

**test-pattern clear**
**test-pattern errors**
**test-pattern insert**
**test-pattern ones**
**test-pattern p215**
**test-pattern p223**
**test-pattern zeros**

## Syntax Description

| | |
|---|---|
| **clear** | Clears the test pattern error count. |
| **errors** | Displays the test pattern error count. |
| **insert** | Inserts an error into the currently active test pattern. |
| **ones** | Generates a test pattern of continous ones. |
| **p215** | Generates a pseudorandom test pattern sequence based on a 15-bit shift register. |
| **p223** | Generates a pseudorandom test pattern sequence based on a 23-bit shift register. |
| **zeros** | Generates a test pattern of continous zeros. |

## Default Value

No default value is necessary for this command.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Usage Examples

The following example enables a 2^15 test pattern:

(config)**#interface t3 1/1**
(config-t3 1/1)**#test-pattern 2^15**

# ATM INTERFACE CONFIG COMMAND SET

To activate the ATM Interface Configuration mode, enter the **interface atm** command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#interface atm 1
(config-atm 1)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:



All other commands for this command set are described in this section in alphabetical order.



# snmp trap

Use the **snmp trap** command to enable all supported Simple Network Management Protocol (SNMP) traps on the interface. Use the **no** form of this command to disable this trap.

## Syntax Description

No subcommands.

## Default Values

By default, all interfaces (except virtual Frame Relay interfaces and sub-interfaces) have SNMP traps enabled.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 3.1 | Command was extended to the SHDSL interface. |
| Release 5.1 | Command was expanded to include Ethernet sub-interfaces and Gigabit Ethernet interfaces. |
| Release 8.1 | Command was expanded to include ATM interfaces. |

## Usage Examples

The following example enables SNMP on the ATM interface:

(config)**#interface atm 1**
(config-atm 1)**#snmp trap**

# snmp trap link-status

Use the **snmp trap link-status** command to control the Simple Network Management Protocol (SNMP) variable ifLinkUpDownTrapEnable (RFC2863) to enable (or disable) the interface to send SNMP traps when there is an interface status change. Use the **no** form of this command to disable this trap.

## Syntax Description

No subcommands.

## Default Values

By default, the ifLinkUpDownTrapEnable OID is enabled for all interfaces except virtual Frame Relay interfaces.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 3.1 | Command was extended to the SHDSL interface. |
| Release 5.1 | Command was expanded to include Gigabit-Ethernet, port-channel, VLAN, E1, and G.703 interfaces. |
| Release 8.1 | Command was expanded to include ATM interfaces. |

## Functional Notes

The **snmp trap link-status** command is used to control the RFC2863 ifLinkUpDownTrapEnable OID (OID number 1.3.6.1.2.1.31.1.1.1.14.0).

## Usage Examples

The following example disables the link-status trap on the ATM interface:

(config)**#interface atm 1**
(config-atm 1)**#no snmp trap link-status**

# ATM SUB-INTERFACE CONFIG COMMAND SET

To activate the ATM Interface Configuration mode, enter the **interface atm** command (and specify a sub-interface) at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#interface atm 1.1
(config-atm 1.1)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:



All other commands for this command set are described in this section in alphabetical order.



# access-policy *<name>*

Use the **access-policy** command to assign a specified access policy to an interface. Access policies are applied to traffic entering an interface. Use the **no** form of this command to remove an access policy association. For more information on using access policies, refer to *ip policy-class <name>* on page 549.

*Configured access policies will only be active if the **ip firewall** command has been entered at the Global Configuration mode prompt to enable the AOS security features. All configuration parameters are valid, but no security data processing will be attempted unless the security features are enabled.*

## Syntax Description

| | |
|---|---|
| *<name>* | Identifies the configured access policy by alphanumeric descriptor (all access policy descriptors are case-sensitive). |

## Default Values

By default, there are no configured access policies associated with an interface.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include Ethernet sub-interfaces. |
| Release 6.1 | Command was expanded to include VLAN interfaces. |

## Functional Notes

To assign an access policy to an interface, enter the interface configuration mode for the desired interface and enter **access-policy** *<policy name>*.

## Usage Examples

The following example associates the access policy **Private** (to allow inbound traffic to the Web server) to the ATM interface 1.1:

Enable the AOS security features:
(config)**#ip firewall**

Associate the access policy with the ATM interface 1.1:

(config)**#interface atm 1.1**
(config-atm 1.1)**#access-policy Private**

# atm routed-bridged ip

Use the **atm routed-bridged ip** command to enable IP routed bridge encapsulation (RBE) on an interface. Use the **no** form of this command to disable RBE operation.

## Syntax Description

No subcommands.

## Default Values

By default, routed bridge encapsulation is disabled.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Usage Examples

The following example enables routed bridge encapsulation:

(config)**#interface atm 1.1**
(config-atm 1.1)#a**tm routed-bridged ip**

# bandwidth *<value>*

Use the **bandwidth** command to provide the bandwidth value of an interface to the higher-level protocols. This value is used in cost calculations. Use the **no** form of this command to restore the default values.

## Syntax Description

| | |
|---|---|
| *<value>* | Specifies bandwidth in kbps. Range is 1 to 4,294,967,295 kbps. |

## Default Values

To view default values use the **show interfaces** command.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |
| Release 8.1 | ATM sub-interface was added. |

## Functional Notes

The **bandwidth** command is an informational value that is communicated to the higher-level protocols to be used in cost calculations. This is a routing parameter only and does not affect the physical interface.

## Usage Examples

The following example sets bandwidth of the ATM sub-interface to 10 Mbps:

(config)**#interface atm 1.1**
(config-atm 1.1)**#bandwidth 10000**

# bridge-group *<value>*

Use the **bridge-group** command to assign an interface to the specified bridge group. Use the **no** form of this command to remove the interface from the bridge group.

## Syntax Description

| | |
|---|---|
| *<value>* | Specifies the bridge group (by number) to which to assign this interface. Range is 1 to 255. |

## Default Values

By default, there are no configured bridge groups.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 8.1 | ATM sub-interface was added. |

## Functional Notes

A bridged network can provide excellent traffic management to reduce collisions and limit the amount of bandwidth wasted with unnecessary transmissions when routing is not necessary. Any two interfaces can be bridged (Ethernet to T1, Ethernet to Frame Relay sub-interface).

## Usage Examples

The following example assigns the ATM sub-interface labeled 1.1 to bridge group 1:

(config)**#interface atm 1.1**
(config-atm 1.1)**#bridge-group 1**

# crypto map *<name>*

Use the **crypto map** command to associate crypto maps with the interface. Use the **no** form of this command to remove a crypto map from an interface.

> *When you apply a map to an interface, you are applying all crypto maps with the given map name. This allows you to apply multiple crypto maps if you have created maps which share the same name but have different map index numbers.*

> *For VPN configuration example scripts, refer to the technical support note **Configuring VPN** located on the ADTRAN OS System Documentation CD provided with your unit.*

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies the crypto map name that you wish to assign to the interface. |

## Default Values

By default, no crypto maps are assigned to an interface.

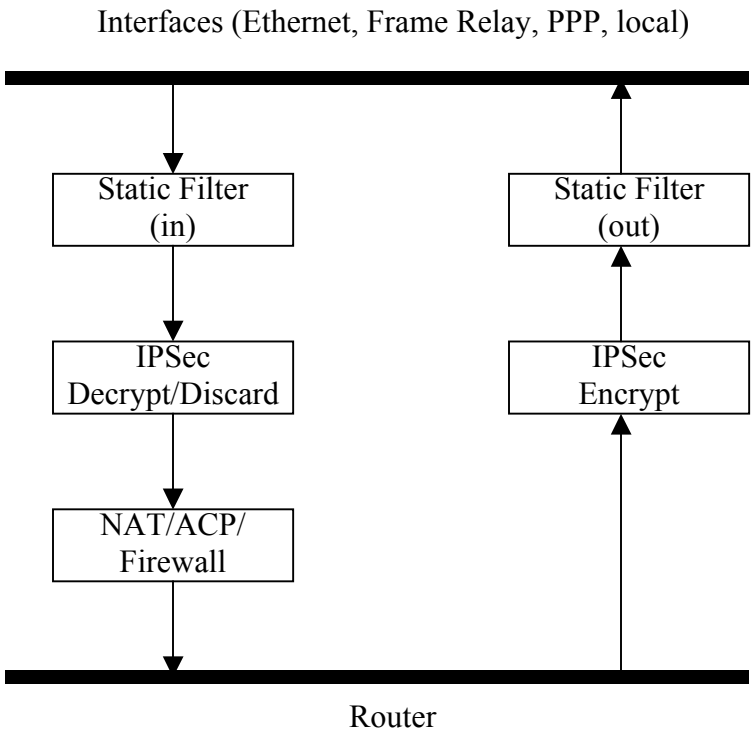
## Command History

| | |
|---|---|
| Release 4.1 | Command was introduced. |
| Release 8.1 | ATM sub-interface was added. |

## Functional Notes

When configuring a system to use both the stateful inspection firewall and IKE negotiation for VPN, keep the following notes in mind.

When defining the policy class and associated access-control lists (ACLs) that describe the behavior of the firewall, do not forget to include the traffic coming into the system over a VPN tunnel terminated by the system. The firewall should be set up with respect to the unencrypted traffic that is destined to be sent or received over the VPN tunnel. The following diagram represents typical AOS data-flow logic.




As shown in the diagram above, data coming into the product is first processed by the static filter associated with the interface on which the data is received. This access group is a true static filter and is available for use regardless of whether the firewall is enabled or disabled. Next (if the data is encrypted) it is sent to the IPSec engine for decryption. The decrypted data is then processed by the stateful inspection firewall. Therefore, given a terminating VPN tunnel, only unencrypted data is processed by the firewall.

The ACLs for a crypto map on an interface work in reverse logic to the ACLs for a policy class on an interface. When specifying the ACLs for a crypto map, the source information is the private local side, unencrypted source of the data. The destination information will be the far end, unencrypted destination of the data. However, ACLs for a policy class work in reverse. The source information for the ACL in a policy class is the far end. The destination information is the local side.

## Usage Examples

The following example applies all crypto maps with the name **MyMap** to the ATM sub-interface:

(config)**#interface atm 1.1**
(config-atm 1.1)**#crypto map MyMap**

# dial-backup auto-backup

Use the **dial-backup auto-backup** command to configure the sub-interface to automatically attempt a dial-backup upon failure. Use the **no** form of this command to disable this feature. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of the command *dial-backup call-mode* on page 1022.

## Syntax Description

No subcommands.

## Default Values

By default, all backup endpoints will automatically attempt dial-backup upon a failure.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 8.1 | ATM sub-interface was added. |

## Usage Examples

The following example enables automatic dial-backup on the endpoint:

(config)**#interface atm 1.1**
(config-atm 1.1)**#dial-backup auto-backup**

# dial-backup auto-restore

Use the **dial-backup auto-restore** command to configure the sub-interface to automatically discontinue dial backup when all network conditions are operational. Use the **no** form of this command to disable the auto-restore feature. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of the command *dial-backup call-mode* on page 1022.

## Syntax Description

No subcommands.

## Default Values

By default, all backup endpoints will automatically restore the primary connection when the failure condition clears.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 8.1 | ATM sub-interface was added. |

## Usage Examples

The following configures the AOS to automatically restore the primary connection when the failure condition clears:

(config)**#interface atm 1.1**
(config-atm 1.1)**#dial-backup auto-restore**

# dial-backup backup-delay *<value>*

Use the **dial-backup backup-delay** command to configure the amount of time the router will wait after the failure condition is recognized before attempting to backup the link. Use the **no** form of this command to return to the default value. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of the command *dial-backup call-mode* on page 1022.

## Syntax Description

| | |
|---|---|
| *<value>* | Specifies the delay period (in seconds) a failure must be active before the AOS will enter backup operation on the interface. Valid range is 10 to 86,400 seconds. |

## Default Values

By default, the **dial-backup backup-delay** period is set to 10 seconds.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 8.1 | ATM sub-interface was added. |

## Usage Examples

The following example configures the AOS to wait 60 seconds (on an endpoint with an active alarm condition) before attempting dial-backup operation:

(config)**#interface atm 1.1**
(config-atm 1.1)**#dial-backup backup-delay 60**

# dial-backup call-mode

Use the **dial-backup call-mode** command to specify whether the configured backup interface answers or originates (or a combination of both ) backup calls. Use the **no** form of this command to return to the default value. Variations of this command include:

**dial-backup call-mode answer**
**dial-backup call-mode answer-always**
**dial-backup call-mode originate**
**dial-backup call-mode originate-answer**
**dial-backup call-mode originate-answer-always**

## Syntax Description

| | |
|---|---|
| **answer** | Answers and backs up primary link on failure. |
| **answer-always** | Answers and backs up regardless of primary link state. |
| **originate** | Originates backup call on primary link failure. |
| **originate-answer** | Originates or answers call on primary link failure. |
| **originate-answer-always** | Originates on failure; answers and backs up always. |

## Default Values

By default, the **dial-backup call-mode** is set to **originate-answer**.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 8.1 | ATM sub-interface was added. |

## Functional Notes

The majority of the configuration for AOS dial-backup implementation is configured via the dial-backup PPP interface configuration commands. However, the numbers dialed are configured in the primary interface. Full sample configurations follow:

**Sample configuration for remote router (dialing out)**

```
hostname “Remote3200”
enable password adtran
!
interface eth 0/1
 ip address 192.168.1.254 255.255.255.0
 no shutdown
!
interface modem 1/3
 no shutdown
!
interface t1 1/1
```

```
 coding b8zs
 framing esf
 clock source line
 tdm-group 1 timeslots 1-24
 no shutdown
!
interface fr 1 point-to-point
 frame-relay lmi-type ansi
 no shutdown
 cross-connect 1 t1 1/1 1 fr 1
!
interface fr 1.16 point-to-point
 frame-relay interface-dlci 16
 ip address 10.1.1.2 255.255.255.252
 dial-backup call-mode originate
 dial-backup number 5551111 analog ppp1
 dial-backup number 5552222 analog ppp1
  no shutdown
!
interface ppp 1
 ip address 172.22.56.1 255.255.255.252
 ppp authentication chap
 username remoter outer password remoteness
 ppp chap hostname local router
 ppp chap password adtran
 no shutdown
!
ip route 192.168.2.0 255.255.255.0 172.22.56.2 255.255.255.252
!
line telnet 0 4
 password password
```

**Sample configuration for central router (dialing in)**

```
hostname “Central3200”
enable password adtran
!
interface eth 0/1
 ip address 192.168.100.254 255.255.255.0
 no shutdown
!
interface modem 1/3
 no shutdown
!
interface t1 1/1
```

```
 coding b8zs
 framing esf
 clock source line
 tdm-group 1 timeslots 1-24
 no shutdown
!
interface fr 1 point-to-point
 frame-relay lmi-type ansi
 no shutdown
 cross-connect 1 t1 1/1 1 fr 1
!
interface fr 1.100 point-to-point
 frame-relay interface-dlci 100
 ip address 10.1.1.1 255.255.255.252
 dial-backup call-mode answer
 dial-backup number 555-8888 analog ppp 1
!
interface ppp 1
 ip address 172.22.56.2 255.255.255.252
 ppp authentication chap
 username local router password adtran
 ppp chap hostname remote router
 ppp chap password remoteness
 no shutdown
!
ip route 192.168.1.0 255.255.255.0 172.22.56.1 255.255.255.252

line telnet 0 4
 password password
```

## Usage Examples

The following example configures AOS to generate backup calls for this endpoint using an analog modem interface (to phone number 555 1111) but never answer calls and specifies **ppp 2** as the backup interface:

```
(config)#interface atm 1.1
(config-atm 1.1)#backup call-mode originate
(config-atm 1.1)#backup number 555 1111 analog ppp 2
```

## Technology Review

This technology review provides information regarding specific dial-backup router behavior (i.e., when the router will perform dial-backup, where in the configuration the AOS accesses specific routing information, etc.):

**Dialing Out**

1. The AOS determines to place an outbound call when either the Layer 1 or Layer 2 has a failure.
2. When placing outbound calls, the AOS matches the number dialed to a PPP interface. This is accomplished with an addition to the **dial-backup number** command (refer to *dial-backup number* on page 1029).
3. When placing the call, the AOS uses the configuration of the related PPP interface for authentication and IP negotiation.
4. If the call fails to connect on the first number dialed, the AOS places a call to the second number (if a second number is configured). The second number to be dialed references a separate PPP interface.

**Dialing In**

1. The AOS receives an inbound call on a physical interface.
2. Caller ID is used to match the **dial-backup number** command to the configured PPP interface.
3. If a match is found, the call connects and the AOS pulls down the primary connection if it is not already in a down state.
4. If no match is found from caller ID, the call is terminated.

# dial-backup connect-timeout <*value*>

Use the **dial-backup connect-timeout** command to specify the number of seconds to wait for a connection after a call is attempted before trying to call again or dialing a different number. It is recommended this number be greater than 60. Use the **no** form of this command to return to the default setting. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of the command *dial-backup call-mode* on page 1022.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the amount of time (in seconds) that the router will wait for a connection before attempting another call. Valid range is 10 to 300 seconds. |

## Default Values

By default, the **dial-backup connect-timeout** period is set to 60 seconds.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 8.1 | ATM sub-interface was added. |

## Usage Examples

The following example configures the AOS to wait 120 seconds before retrying a failed dial-backup call:

(config)**#interface atm 1.1**
(config-atm 1.1)**#dial-backup connect-timeout 120**

# dial-backup force

Use the **dial-backup force** command to manually override the automatic dial-backup feature. This can be used to force a link into backup to allow maintenance to be performed on the primary link without disrupting data. Use the **no** form of this command to return to the normal dial-backup operation state. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of the command *dial-backup call-mode* on page 1022. Variations of this command include:

**dial-backup force backup**
**dial-backup force primary**

## Syntax Description

| | |
|---|---|
| **backup** | Forces backup regardless of primary link state. |
| **primary** | Forces primary link regardless of its state. |

## Default Values

By default, this feature is disabled.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 8.1 | ATM sub-interface was added. |

## Usage Examples

The following example configures the AOS to force this endpoint into dial-backup:

(config)**#interface atm 1.1**
(config-atm 1.1)**#dial-backup force backup**

# dial-backup maximum-retry <*value*>

Use the **dial-backup maximum-retry** command to select the number of calls the router will make when attempting to backup a link. Use the **no** form of this command to return to the default state. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of the command *dial-backup call-mode* on page 1022.

## Syntax Description

| | |
|---|---|
| <*value*> | Selects the number of call retry attempts that will be made after a link failure. Valid range is 0 to 15 attempts.<br>Setting this value to 0 will allow unlimited retries during the time the network is failed. |

## Default Values

By default, the **dial-backup maximum-retry** period is set to 0 attempts.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 8.1 | ATM sub-interface was added. |

## Usage Examples

The following example configures the AOS to retry a dial-backup call 4 times before considering backup operation not available:

(config)**#interface atm 1.1**
(config-atm 1.1)**#dial-backup maximum-retry 4**

# dial-backup number

Use the **dial-backup number** command to configure the phone number and the call type the router will dial upon network failure. Multiple entries can be made for an interface to allow alternate sites to be dialed. Use the **no** form of this command to disable this feature. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of the command *dial-backup call-mode* on page 1022. Variations of this command include:

**dial-backup number** <*number*> **analog ppp** <*interface*>
**dial-backup number** <*number*> **digital-56k** <*isdn min chan*> <*isdn max chan*> **ppp** <*interface*>
**dial-backup number** <*number*> **digital 64k** <*isdn min chan*> <*isdn max chan*> **ppp** <*interface*>

## Syntax Description

| | |
|---|---|
| <*number*> | Specifies the phone numbers to call when the backup is initiated. |
| **analog ppp** | Indicates the number connects to an analog modem. |
| **digital-56k** | Indicates the number belongs to a digital 56 kbps per DS0 connection. |
| **digital-64k** | Indicates the number belongs to a digital 64 kbps per DS0 connection. |
| <*isdn min chan*> | Specifies the minimum number of DS0s required for a digital 56 or 64 kbps connection. Range is 1 to 24 DS0s. |
| <*isdn max chan*> | Specifies the maximum number of DS0s desired for a digital 56 or 64 kbps connection. Range is 1 to 24 DS0s. |
| **ppp** <*interface*> | Specifies the PPP interface to use as the backup for this interface. For example, **ppp 1**. |

## Default Values

By default, there are no configured dial backup numbers.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 8.1 | ATM sub-interface was added. |

## Usage Examples

The following example configures the AOS to dial 704-555-1212 (digital 64 kbps connection) to initiate dial-backup operation for this endpoint using the configured **ppp 1** backup interface:

(config)**#interface atm 1.1**
(config-atm 1.1)**#dial-backup number 7045551212 digital-64k 1 1 ppp 1**

# dial-backup priority <*value*>

Use the **dial-backup priority** command to select the backup priority for this interface. This allows the user to establish the highest priority backup link and ensure that link will override backups attempted by lower priority links. Use the **no** form of this command to return to the default value. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of the command *dial-backup call-mode* on page 1022.

## Syntax Description

| | |
|---|---|
| <*value*> | Sets the relative priority to this link. Valid range is 0 to 100. A value of 100 designates the highest priority. |

## Default Values

By default, the **dial-backup priority** is set to 50.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 8.1 | ATM sub-interface was added. |

## Usage Examples

The following example assigns the highest priority to this endpoint:

(config)**#interface atm 1.1**
(config-atm 1.1)**#dial-backup priority 100**

# dial-backup randomize-timers

Use the **dial-backup randomize-timers** command to randomize the call timers to minimize potential contention for resources. Use the **no** form of this command to return to the default value. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of the command *dial-backup call-mode* on page 1022.

## Syntax Description

No subcommands.

## Default Values

By default, the AOS does not randomize the dial-backup call timers.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 8.1 | ATM sub-interface was added. |

## Usage Examples

The following example configures the AOS to randomize the dial-backup timers associated with this endpoint:

(config)**#interface atm 1.1**
(config-atm 1.1)**#dial-backup randomize-timers**

# dial-backup redial-delay *<value>*

Use the **dial-backup redial-delay** command to configure the delay after an unsuccessful call until the call will be re-tried. Use the **no** form of this command to return to the default setting. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of the command *dial-backup call-mode* on page 1022.

## Syntax Description

| | |
|---|---|
| *<value>* | Specifies the delay (in seconds) between attempting to redial a failed backup attempt. Valid range is 10 to 3600 seconds. |

## Default Values

By default, the **dial-backup redial-delay** period is set to 10 seconds.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 8.1 | ATM sub-interface was added. |

## Usage Examples

The following example configures a redial delay of **25** seconds on this endpoint:

(config)**#interface atm 1.1**
(config-atm 1.1)**#dial-backup redial-delay 25**

# dial-backup restore-delay *<value>*

Use the **dial-backup restore-delay** command to configure the amount of time the router will wait after the network is restored before disconnecting the backup link and reverting to the primary. This setting is used to prevent disconnecting the backup link if the primary link is "bouncing" in and out of alarm. Use the **no** form of this command to return to the default setting. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of the command *dial-backup call-mode* on page 1022.

## Syntax Description

| | |
|---|---|
| *<value>* | Specifies the number of seconds the AOS will wait (after a primary link is restored) before disconnecting dial-backup operation. Range is 10 to 86,400 seconds. |

## Default Values

By default, the **dial-backup restore-delay** period is set to 10 seconds.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 8.1 | ATM sub-interface was added. |

## Usage Examples

The following example configures the AOS to wait **30** seconds before disconnecting dial-backup operation and restoring the primary connection for this endpoint:

(config)**#interface atm 1.1**
(config-atm 1.1)**#dial-backup restore-delay 30**

# dial-backup schedule

Use the **dial-backup schedule** command to set the time of day that backup will be enabled. Use this command if backup is desired only during normal business hours and on specific days of the week. Use the **no** form of this command to disable dial-backup (as specified). For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of the command *dial-backup call-mode* on page 1022. Variations of this command include:

**dial-backup schedule day** <*name*>
**dial-backup schedule enable-time** <*value*>
**dial-backup schedule disable-time** <*value*>

## Syntax Description

| | |
|---|---|
| **day** <*name*> | Sets the days to allow backup. Valid range is Monday through Sunday. |
| **enable-time** <*value*> | Sets the time of day to enable backup. Time is entered in 24-hour format (00:00). |
| **disable-time** <*value*> | Sets the time of day to disable backup. Time is entered in 24-hour format (00:00). |

Default Values

By default, dial-backup is enabled for all days and times if the dial-backup auto-backup command has been issued and the dial-backup schedule has not been entered.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 8.1 | ATM sub-interface was added. |

## Usage Examples

The following example enables dial-backup Monday through Friday 8:00 am to 7:00 pm:

(config)**#interface atm 1.1**
(config-atm 1.1)**#dial-backup schedule enable-time 08:00**
(config-atm 1.1)**#dial-backup schedule disable-time 19:00**
(config-atm 1.1)**#no dial-backup schedule day Saturday**
(config-atm 1.1)**#no dial-backup schedule day Sunday**

# dial-backup shutdown

Use the **dial-backup shutdown** command to deactivate all dial-backup functionality in the unit. Dial-backup configuration parameters are kept intact, but the unit will not initiate (or respond) to dial-backup sequences in the event of a network outage. Use the **no** form of this command to reactivate the dial-backup interface. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of the command *dial-backup call-mode* on page 1022.

## Syntax Description

No subcommands.

## Default Values

By default, all AOS interfaces are disabled.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 8.1 | ATM sub-interface was added. |

## Usage Examples

The following example deactivates the configured dial-backup interface:

(config)**#interface atm 1.1**
(config-atm 1.1)**#dial-backup shutdown**

# dynamic-dns

Use the **dynamic-dns** command to configure Dynamic DNS service provided by Dynamic Network Services, Inc. (www.dyndns.org). Use the **no** versions of these commands to disable these features. Variations of this command include:

**dynamic-dns custom** <*hostname*> <*minutes*>
**dynamic-dns dyndns** <*hostname*> <*username*> <*password*>
**dynamic-dns dyndns-custom** <*hostname*> <*username*> <*password*>
**dynamic-dns dyndns-static** <*hostname*> <*username*> <*password*>

## Syntax Description

| | |
|---|---|
| <*hostname*> | Specifies the host name for the server that updates the Dynamic Domain Name Server (DNS). |
| <*minutes*> | Specifies the intervals in minutes to update the server with information (updates also occur when the interface's IP address changes regardless of the update intervals).<br>Refer to *Functional Notes* below for additional argument descriptions. |

## Default Values

No default is necessary for this command.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |
| Release 12.1 | Command was expanded. |

## Functional Notes

**custom** - Constanttime.com's Custom Dynamic DNS$^{SM}$ service allows you complete access and management control over your domain name regardless of where you purchased/registered it. This allows to manage IP address mappings (A records), domain aliases (CNAME records) and mail servers (MX records).

**dyndns** - The Dynamic DNS$^{SM}$ offered by Dynamic Network Services Inc., (DynDNS.org) allows you to alias a dynamic IP address to a static host name in various domains. This allows your unit to be more easily accessed from various locations on the Internet. This service is provided for up to five host names.

**dyndns-custom -** DynDNS.org's Custom DNS$^{SM}$ service provides a full DNS solution, giving you complete control over an entire domain name. A web-based interface provides two levels of control over your domain, catering to average or advanced users. Five globally redundant DNS servers ensure that your domain will always resolve.

A choice of two interfaces is available. The basic interface is designed for most users. It comes preconfigured for the most common configuration and allows for easy creation of most common record types. The advanced interface is designed for system administrators with a solid DNS background, and provides layout and functionality similar to a BIND zone file allowing for the creation of nearly any record type.

Custom DNS$^{SM}$ can be used with both static and dynamic IPs and has the same automatic update capability through Custom DNS-aware clients as Dynamic DNS.

**dyndns-static -** The Static DNS service is similar to DynDNS.org's Dynamic DNS$^{SM}$ service in that it allows a host name such as yourname.dyndns.org to point to your IP address. Unlike a Dynamic DNS host, a Static DNS host does not expire after 35 days without updates, but updates take longer to propagate though the DNS system. This service is provided for up to five host names.

If your IP address does not change often or at all but you still want an easy name to remember it by (without having to purchase your own domain name), Static DNS service is ideal for you.

If you would like to use your own domain name (such as yourname.com), you need Custom DNS service which also provides full dynamic and static IP address support.

**Usage Examples**

The following example sets the dynamic-dns to **dyndns-custom** with host name **host**, user name **user**, and password **pass**:

(config)**#interface atm 1.1**
(config-atm 1.1)**#dynamic-dns dyndns-custom host user pass**

# encapsulation

Use the **encapsulation** command to configure the encapsulation type for the ATM Adaption Layer (AAL) of the ATM Protocol Reference Model. Use the **no** form of this command to return to the default setting. Variations of this command include:

**encapsulation aal5mux ip**
**encapsulation aal5mux ppp**
**encapsulation aaal5snap**

## Syntax Description

| | |
|---|---|
| **aal5mux ip** | Specifies encapsulation type for multiplexed virtual circuits using the IP protocol. |
| **aal5mux ppp** | Specifies encapsulation type for multiplexed virtual circuits using the Point-to-Point (PPP) protocol. |
| **aal5snap** | Specifies encapsulation type that supports LLC/SNAP protocols. |

## Default Values

By default, the encapsulation type is **aal5snap**.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Functional Notes

For PPP and PPoE, the encapsulation type can be **aal5snap** or **aal5mux ppp**. For IP with no bridging, the encapsulation type can be **aal5snap** or **aal5mux ip**. For IP with bridging, the encapsulation type can only be **aal5snap**. For bridging, the encapsulation type can only be **aal5snap**.

## Usage Examples

The following example sets the encapsulation type to **all5snap**:

(config)**#interface atm 1.1**
(config-atm 1.1)**#encapsulation aal5snap**

# fair-queue

Use the **fair-queue** command to enable weighted fair queuing (WFQ) on an interface. Use the **no** form of this command to disable WFQ and enable first-in-first-out (FIFO) queueing for an interface. WFQ is enabled by default for WAN interfaces. Variations of this command include:

**fair-queue**
**fair-queue** <*value*>

## Syntax Description

| | |
|---|---|
| <*value*> | Optional. Value that specifies the maximum number of packets that can be present in each conversation sub-queue. Packets received for a conversation after this limit is reached are discarded. Range is 16 to 512 packets. |

## Default Values

By default, **fair-queue** is enabled with a threshold of 64 packets.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |
| Release 8.1 | ATM sub-interface was added. |

## Usage Examples

The following example enables WFQ on the interface with a threshold set at 100 packets:

(config)**#interface atm 1.1**
(config-atm 1.1)**#fair-queue 100**

# hold-queue <*value*> out

Use the **hold-queue out** command to change the overall size of an interface's WAN output queue. Use the **no** form of this command to return to the default settings.

## Syntax Description

| | |
|---|---|
| <*value*> | The total number of packets the output queue can contain before packets are dropped. Range is 16 to 1000 packets. |

## Default Values

The default queue size for WFQ is 400. The default queue size for PPP FIFO and Frame Relay round-robin is 200.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |
| Release 8.1 | ATM sub-interface was added. |

## Usage Examples

The following example sets the overall output queue size to 700:

(config)**#interface atm 1.1**
(config-atm 1.1)**#hold-queue 700 out**

# ip access-group *<name>*

Use the **ip access-group** command to create an access control list to be used for packets transmitted on or received from the specified interface. Use the **no** form of this command to disable this type of control. Variations of this command include:

**ip access-group** <*name*> **in**
**ip access-group** <*name*> **out**

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies the assigned IP access control list name. |
| **in** | Enables access control on packets received on the specified interface. |
| **out** | Enables access control on packets transmitted on the specified interface. |

## Default Values

By default, these commands are disabled.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |
| Release 8.1 | ATM sub-interface was added. |

## Functional Notes

When this command is enabled, the IP destination address of each packet must be validated before being passed through. If the packet is not acceptable per these settings, it is dropped.

## Usage Examples

The following example sets up the router to only allow Telnet traffic into the ATM sub-interface:

(config)**#ip access-list extended TelnetOnly**
(config-ext-nacl)**#permit tcp any any eq telnet**
(config-ext-nacl)**#interface atm 1.1**
(config-atm 1.1)**#ip access-group TelnetOnly in**

# ip address dhcp

Use the **ip address dhcp** command to use Dynamic Host Configuration Protocol (DHCP) to obtain an address on the interface. Use the **no** form of this command to remove a configured IP address (using DHCP) and disable DHCP operation on the interface. Variables that may be used with this command to further define the DHCP configuration include:

**ip address dhcp client-id [**<*interface*> **|** <*identifier*>**] [hostname** <*"string"*>**] [track** <*name*>**]**
**[**<*administrative distance*>**]**
**ip address dhcp hostname** <*"string"*> **[no-default-route | no-domain-name | no-nameservers] [track** <*name*>**] [**<*administrative distance*>**]**
**ip address dhcp [no-default-route | no-domain-name | no-nameservers] [track** <*name*>**]**
**[**<*administrative distance*>**]**
**ip address dhcp track** <*name*> **[**<*administrative distance*>**]**

## Syntax Description

| | |
|---|---|
| <*administrative distance*> | Optional. Specifies the administrative distance to use when adding the DHCP gateway into the route table. It is used to determine the best route when multiple routes to the same destination exist. The smaller the administrative distance the more reliable the route. Range is 1 to 255. |
| **client-id** | Optional. Specifies the client identifier used when obtaining an IP address from a DHCP server. |
| <*interface*> | Specifies an interface, thus defining the client identifier as the hexadecimal MAC address of the specified interface (including a hexadecimal number added to the front of the MAC address to identify the media type).<br>For example, specifying the **client-id ethernet 0/1** (where the Ethernet interface has a MAC address of d217.0491.1150) defines the client identifier as **01:d2:17:04:91:11:50** (where 01 defines the media type as Ethernet). Refer to *hardware-address* <*mac address*> on page 1973 for a detailed listing of media types. |
| <*identifier*> | Specifies a custom client-identifier using a text string (that is converted to a hexadecimal equivalent) or 7 to 28 hexadecimal numbers (with colon delimiters).<br>For example, a custom client identifier of **0f:ff:ff:ff:ff:51:04:99:a1** may be entered using the <*identifier*> option. |
| **hostname** <*"string"*> | Optional. Specifies a text string (to override the global router name) to use as the name in the DHCP option 12 field. The string is enclosed in quotation marks and can consist of up to 35 characters. |
| **no-default-route** | Optional. Specifies that no default route is obtained via DHCP. |
| **no-domain-name** | Optional. Specifies that no domain name is obtained via DHCP. |
| **no-nameservers** | Optional. Specifies that no DNS servers are obtained via DHCP. |
| **track** <*name*> | Optional. Attaches a network monitoring track to the DHCP client. The DHCP gateway route for this client will only reside in the route table while the track is in the pass state. For more information on configuring track objects, refer to *track* <*name*> on page 691. |

## Default Values

| | |
|---|---|
| *<administrative distance>* | By default, the administrative distance value is 1. |
| **client-id** | Optional. By default, the client identifier is populated using the following formula:<br>TYPE: INTERFACE SPECIFIC INFO : MAC ADDRESS<br>Where TYPE specifies the media type in the form of one hexadecimal byte (refer to *hardware-address <mac address>* on page 1973 for a detailed listing of media types), and the MAC ADDRESS is the Media Access Control (MAC) address assigned to the first Ethernet interface in the unit in the form of six hexadecimal bytes. (For units with a single Ethernet interface, the MAC ADDRESS assigned to Ethernet 0/1 is used in this field).<br>INTERFACE SPECIFIC INFO is only used for Frame Relay interfaces and can be determined using the following:<br>FR_PORT#: Q.922 ADDRESS<br>Where the FR_PORT# specifies the label assigned to the virtual Frame Relay interface using four hexadecimal bytes. For example, a virtual Frame Relay interface labeled 1 would have a FR_PORT# of 00:00:00:01.<br>The Q.922 ADDRESS field is populated using the following: |

| 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
|---|---|---|---|---|---|---|---|
| DLCI (high order) | | | | | | C/R | EA |
| DLCI (lower) | | | | FECN | BECN | DE | EA |

Where the FECN, BECN, C/R, DE, and high order extended address (EA) bits are assumed to be 0 and the lower order EA bit is set to 1.

The following list provides a few example DLCIs and associated Q.922 address:

DLCI (decimal) / Q.922 address (hex)

16 / 0x0401
50 / 0x0C21
60 / 0x0CC1
70 / 0x1061
80 / 0x1401

| | |
|---|---|
| **hostname** <*"string"*> | By default, the host name is the name configured using the Global Configuration **hostname** command. |

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |
| Release 8.1 | Command was expanded to include ATM sub-interface. |
| Release 13.1 | Command was expanded to include track and administrative distance. |

## Functional Notes

DHCP allows interfaces to acquire a dynamically assigned IP address from a configured DHCP server on the network. Many Internet Service Providers (ISPs) require the use of DHCP when connecting to their services. Using DHCP reduces the number of dedicated IP addresses the ISP must obtain. Consult your ISP to determine the proper values for the **client-id** and **hostname** fields.

## Usage Examples

The following example enables DHCP operation on the ATM sub-interface 1.1:

(config)**#interface atm 1.1**
(config-atm 1.1)**#ip address dhcp**

The following example enables DHCP operation on the ATM sub-interface 1.1 utilizing hostname **adtran** and does not allow obtaining a default route, domain name, or nameservers. It also sets the administrative distance as **5**:

(config)**#interface atm 1.1**
(config-atm 1.1)**#ip address dhcp hostname “adtran” no-default-route no-domain-name no-nameservers 5**

# ip address *<ip address> <subnet mask>*

Use the **ip address** command to define an IP address on the specified interface. Use the optional **secondary** keyword to define a secondary IP address. Use the **no** form of this command to remove a configured IP address. Variations of this command include:

**ip address** *<ip address> <subnet mask>*
**ip address** *<ip address> <subnet mask>* **secondary**

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies a valid IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| *<subnet mask>* | Specifies the subnet mask that corresponds to a range of IP addresses (network) or a specific host. Subnet masks can be expressed in dotted decimal notation (for example, **255.255.255.0**) or as a prefix length (for example, **/24**). |
| **secondary** | Optional. Configures a secondary IP address for the specified interface. |

## Default Values

By default, there are no assigned IP addresses.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 8.1 | ATM sub-interface was added. |

## Functional Notes

Use secondary IP addresses to allow dual subnets on a single interface (when you need more IP addresses than the primary subnet can provide). When using secondary IP addresses, avoid routing loops by verifying that all devices on the network segment are configured with secondary IP addresses on the secondary subnet.

## Usage Examples

The following example configures a secondary IP address of **192.22.72.101/30**:

(config)**#interface atm 1.1**
(config-atm 1.1)**#ip address 192.22.72.101 255.255.255.252 secondary**

# ip dhcp

Use the **ip dhcp** command to release or renew the DHCP IP address. This command is only applicable when using DHCP for IP address assignment. Variations of this command include:

**ip dhcp release**
**ip dhcp renew**

## Syntax Description

| | |
|---|---|
| **release** | Releases DHCP IP address. |
| **renew** | Renews DHCP IP address. |

## Default Values

No default values required for this command.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |
| Release 8.1 | ATM sub-interface was added. |

## Usage Examples

The following example releases the IP DHCP address for the ATM sub-interface 1.1:

(config)**#interface atm 1.1**
(config-atm 1.1)**#ip dhcp release**

# ip ffe

Use the **ip ffe** command to enable the FastFlow Engine (FFE) on this interface with the default number of entries. Use the **no** form of these commands to disable the FFE. Variations of this command include:

**ip ffe max-entries** <*entries*>

*Issuing this command will cause all FFE entries on this interface to be cleared.*

## Syntax Description

| | |
|---|---|
| **max-entries** <*entries*> | Specifies the maximum number of entries stored in the flow table. Valid range is from 1 to 8192. |

## Default Values

By default, the FFE is disabled. The default number of **max-entries** is 4096.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Functional Notes

The FFE can be used to help reduce routing overhead and thus reduce overall routing times. Routing times are reduced by the creation of a flow table on the ingress interface. The maximum number of entries that can be stored in the flow table at any one time may be specified by using the **max-entries** subcommand. When features such as firewall, VPN, and static filters are enabled, routing time reduction may not be realized.

## Usage Examples

The following example enables the FFE and sets the maximum number of entries in the flow table to 50:

(config)**#interface atm 1.1**
(config-atm 1.1)**#ip ffe max-entries 50**

## Technology Review

The FastFlow system goal is to increase IP packet throughput by moving as much of the packet processing into the engine as possible. Packets are classified into flows based upon the IP protocol (TCP, UDP, ICMP, etc.), the source and destination IP addresses, IP TOS, and the protocol-specific information such as the source and destination port numbers. Flows are defined as the unidirectional representation of a conversation between two IP hosts. Each ingress interface keeps its own flow table, a collection of flow entries.

The first packet in a flow that is forwarded through the unit will build a flow entry. When a flow entry is looked up but no entry is found, a FastFlowBuilder object is allocated and attached to the packet. As the packet passes through the various processing layers, each subsystem will add processing to the FastFlowBuilder. When packet is about to be forwarded out of the egress interface, the FastFlowBuilder will be finalized. That is, the flow entry being built will be checked for completeness and committed to the flow table on the ingress interface. Subsequent flow matches can then bypass the normal processing layers.

# ip helper-address *<ip address>*

Use the **ip helper-address** command to configure the AOS to forward User Datagram Protocol (UDP) broadcast packets received on the interface. Use the **no** form of this command to disable forwarding packets.

> NOTE *The **ip helper** command must be used in conjunction with the **ip forward-protocol** command to configure the AOS to forward UDP broadcast packets. See ip forward-protocol udp <value> on page 532 for more information.*

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies the destination IP address for the forwarded UDP packets. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

## Default Values

By default, broadcast UDP packets are not forwarded.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 8.1 | ATM sub-interface was added. |

## Functional Notes

When used in conjunction with the **ip forward-protocol** command, the **ip helper-address** feature allows you to customize which broadcast packets are forwarded.

To implement the helper address feature, assign an address(es) (specifying the device that needs to receive the broadcast traffic) to the interface closest to the host that transmits the broadcast packets. When broadcast packets (of the specified type forwarded using the **ip forward-protocol** command) are received on the interface, they will be forwarded to the device that needs the information.

Only packets meeting the following criteria are considered eligible by the **ip helper-address** feature:

1. The packet IP protocol is UDP.
2. Any UDP port specified using the **ip forward-protocol** command.
3. The media access control (MAC) address of the frame is an all-ones broadcast address (ffff.ffff.ffff).
4. The destination IP address is broadcast defined by all ones (255.255.255.255) or a subnet broadcast (for example, 192.33.4.251 for the 192.33.4.248/30 subnet).

## Usage Examples

The following example forwards all DNS broadcast traffic to the DNS server with IP address 192.33.5.99:

(config)**#ip forward-protocol udp domain**
(config)**#interface atm 1.1**
(config-atm 1.1)**#ip helper-address 192.33.5.99**

# ip igmp

Use the **ip igmp** command to configure multicasting-related functions for the interface. Variations of this command include:

**ip igmp immediate-leave**
**ip igmp last-member-query-interval** <*milliseconds*>
**ip igmp querier-timeout** <*seconds*>
**ip igmp query-interval** <*seconds*>
**ip igmp query-max-response-time** <*seconds*>
**ip igmp static-group** <*address*>
**ip igmp version [1 | 2]**

## Syntax Description

| | |
|---|---|
| **immediate-leave** | Specifies that if only one host (or IGMP snooping switch) is connected to the interface, when a leave is received, multicast of that group is immediately terminated as opposed to sending a group query and timing out the group if no device responds. Works in conjunction with **ip igmp last-member-query-interval**. Applies to all groups when configured. Use the **no** form of this command to disable the **immediate-leave** feature. |
| **last-member-query-interval** <*milliseconds*> | Controls the timeout (in milliseconds) used to detect whether any group receivers remain on an interface after a receiver leaves a group. If a receiver sends a leave-group message (IGMP Version 2), the router sends a group-specific query on that interface. After twice the time specified by this command plus as much as one second longer, if no receiver responds, the router removes that interface from the group and stops sending that group's multicast packets to the interface. Range is 100 to 65,535 ms. Use the **no** form of this command to return to the default setting. |
| **querier-timeout** <*seconds*> | Specifies the interval (in seconds) that the router waits after the current querier's last query before it takes over as querier (IGMP V2). Range is 60 to 300 seconds. Use the **no** form of this command to return to the default setting. |
| **query-interval** <*seconds*> | Specifies the interval (in seconds) at which IGMP queries are sent on an interface. Host query messages are addressed to the all-hosts multicast group with an IP TTL of 1. The router uses queries to detect whether multicast group members are on the interface and to select an IGMP designated router for the attached segment (if more than one multicast router exists). Only the designated router for the segment sends queries. For IGMP V2, the designated router is the router with the lowest IP address on the segment. Range is 0 to 65,535 seconds. Use the **no** form of this command to return to the default setting. |
| **query-max-response-time** <*seconds*> | Specifies the maximum response time (in seconds) advertised by this interface in queries when using IGMP V2. Hosts are allowed a random time within this period to respond, reducing response bursts. Use the **no** form of this command to return to the default setting. |

## Syntax Description

| | |
|---|---|
| **static-group** <*address*> | Configures the router's interface to be a statically-connected member of the specified group. Packets received on the correct RPF interface are forwarded to this interface regardless of whether any receivers have joined the specified group using IGMP. Use the **no** form of this command to remove a configured static group. |
| **version [1 \| 2]** | Sets the interface's IGMP version. Use the **no** form of this command to return to the default setting. |

## Default Values

| | |
|---|---|
| **ip igmp immediate-leave** | No default |
| **ip igmp last-member-query-interval** | 1000 milliseconds |
| **ip igmp querier-timeout** | 2x the **query-interval** value |
| **ip igmp query-interval** | 60 seconds |
| **ip igmp query-max-response-time** | 10 seconds |
| **ip igmp static-group** | No default |
| **ip igmp version** | Version 1 |

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |
| Release 8.1 | ATM sub-interface was added. |

## Usage Examples

The following example sets the query message interval on the interface to 200 milliseconds:

(config)**#interface atm 1.1**
(config-atm 1.1)**#ip igmp last-member-query-interval 200**

# ip mcast-stub downstream

Use the **ip mcast-stub downstream** command to enable multicast forwarding and IGMP (router mode) on an interface and place it in multicast stub downstream mode. Use the **no** form of this command to disable.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |
| Release 8.1 | ATM sub-interface was added. |

## Functional Notes

This command is used in IP multicast stub applications in conjunction with the **ip mcast-stub helper-address** and **ip mcast-stub upstream** commands. Downstream interfaces connect to segments with multicast hosts. Multiple interfaces may be configured in downstream mode; however, interfaces connecting to the multicast network (upstream) should not be configured in downstream mode. Interfaces configured as downstream should have the lowest IP address of all IGMP-capable routers on the connected segment in order to be selected as the designated router and ensure proper forwarding. Refer to *ip mcast-stub helper-address <ip address>* on page 546 and *ip mcast-stub upstream* on page 1056 for more information.

## Usage Examples

The following example enables multicast forwarding and IGMP on the interface:

(config)**#interface atm 1.1**
(config-atm 1.1)**#ip mcast-stub downstream**

# ip mcast-stub fixed

Use the **ip mcast-stub fixed** command to allow forwarding of multicast traffic on a selected interface after enabling multicast routing. Use the **no** form of this command to disable this mode.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Functional Notes

Multicast routing must be enabled prior to setting **ip mcast-stub fixed** on the selected interface. Also, use the **ip igmp static-group** <*A.B.C.D*> command to receive multicast traffic without host-initiated Internet Group Management Protocol (IGMP) activity on the selected interface. Otherwise, all host-initiated IGMP transactions will enter multicast routes on the router's interface involved with IGMP activities.

### Usage Examples

The following example enables multicast traffic forwarding and IGMP on the interface:

(config)**#interface atm1.1**
(config-atm 1.1)**#ip mcast-stub fixed**

# ip mcast-stub helper-enable

Use the **ip mcast-stub helper-enable** command to assign the **ip mcast-stub helper-address** as the IGMP proxy. Use the **no** form of this command to disable.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Functional Notes

This command is used in IP multicast stub applications in conjunction with the **ip mcast-stub helper-address, ip mcast-stub upstream**, and **ip mcast-stub downstream** commands. When enabled, the interface becomes a helper forwarding interface. The IGMP host function is dynamically enabled and the interface becomes the active upstream interface, enabling the router to perform as an IGMP proxy. Refer to *ip mcast-stub helper-address <ip address>* on page 546, *ip mcast-stub downstream* on page 1053, and *ip mcast-stub upstream* on page 1056 for more information.

## Usage Examples

The following example sets the helper address as the IGMP proxy:

```
(config)#interface atm 1.1
(config-atm 1.1)#ip mcast-stub helper-enable
```

# ip mcast-stub upstream

Use the **ip mcast-stub upstream** command to enable multicast forwarding on an interface and place it in multicast stub upstream mode. Use the **no** form of this command to disable.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |
| Release 8.1 | ATM sub-interface was added. |

## Functional Notes

This command is used in IP multicast stub applications in conjunction with the **ip mcast-stub helper-address** and **ip mcast-stub downstream** commands. When enabled, the interface becomes a candidate to be a helper forwarding interface. If chosen as the best path toward the helper address by the router's unicast route table, the IGMP host function is dynamically enabled and the interface becomes the active upstream interface, enabling the router to perform as an IGMP proxy. Though multiple interfaces may be candidates, no more than one interface will actively serve as the helper forwarding interface. Refer to *ip mcast-stub helper-address <ip address>* on page 546 and *ip mcast-stub downstream* on page 1053 for more information.

## Usage Examples

The following example enables multicast forwarding on the interface:

(config)**#interface atm 1.1**
(config-atm 1.1)**#ip mcast-stub upstream**

# ip ospf

Use the **ip ospf** command to customize OSPF settings (if needed). Use the **no** form of these commands to return to the default settings. Variations of this command include:

**ip ospf authentication-key** <*password*>
**ip ospf cost** <*value*>
**ip ospf dead-interval** <*seconds*>
**ip ospf hello-interval** <*seconds*>
**ip ospf message-digest-key [1 | 2] md5** <*key*>
**ip ospf priority** <*value*>
**ip ospf retransmit-interval** <*seconds*>
**ip ospf transmit-delay** <*seconds*>

## Syntax Description

| | |
|---|---|
| **authentication-key** <*password*> | Assigns a simple-text authentication password to be used by other routers using the OSPF simple password authentication. |
| **cost** <*value*> | Specifies the OSPF cost of sending a packet on the interface. This value overrides any computed cost value. Range is 1 to 65,535. |
| **dead-interval** <*seconds*> | Sets the maximum interval (in seconds) allowed between hello packets. If the maximum is exceeded, neighboring devices will determine that the device is down. Range is 0 to 32,767 seconds. |
| **hello-interval** <*seconds*> | Specifies the interval (in seconds) between hello packets sent on the interface. Range is 0 to 32,767 seconds. |
| **message-digest-key [1 \| 2] md5** <*key*> | Configures OSPF Message Digest 5 (MD5) authentication (16-byte max) keys. |
| **priority** <*value*> | Sets the OSPF priority. The value set in this field helps determine the designated router for this network. Range is 0 to 255. |
| **retransmit-interval** <*seconds*> | Specifies the interval (in seconds) between link-state advertisements (LSAs). Range is 0 to 32,767 seconds. |
| **transmit-delay** <*seconds*> | Sets the estimated time (in seconds) required to send an LSA on the interface. Range is 0 to 32,767 seconds. |

## Default Values

| | |
|---|---|
| **retransmit-interval** <*seconds*> | 5 seconds |
| **transmit-delay** <*seconds*> | 1 second |
| **hello-interval** <*seconds*> | 10 seconds: Ethernet, point-to-point, Frame Relay, and PPP |
| **dead-interval** <*seconds*> | 40 seconds |

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |
| Release 8.1 | ATM sub-interface was added. |

## Usage Examples

The following example sets the maximum number of seconds allowed between hello packets to 25,000:

(config)**#interface atm 1.1**
(config-atm 1.1)**#ip ospf dead-interval 25000**

# ip ospf authentication

Use the **ip ospf authentication** command to authenticate an interface that is performing OSPF authentication. Use the **no** form of this command to return to the default setting. Variations of this command include:

**ip ospf authentication message-digest**
**ip ospf authentication null**

## Syntax Description

| | |
|---|---|
| **message-digest** | Optional. Specifies message-digest authentication type. |
| **null** | Optional. Specifies that no authentication be used. |

## Default Values

By default, this is set to null (meaning no authentication is used).

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |
| Release 8.1 | ATM sub-interface was added. |

## Usage Examples

The following example specifies that no authentication will be used on the ATM sub-interface 1.1:

(config)**#interface atm 1.1**
(config-atm 1.1)**#ip ospf authentication null**

# ip ospf network

Use the **ip ospf network** command to specify the type of network on this interface. Use the **no** form of this command to return to the default setting. Variations of this command include:

**ip ospf network broadcast**
**ip ospf network point-to-point**

## Syntax Description

| | |
|---|---|
| **broadcast** | Sets the network type for broadcast. |
| **point-to-point** | Sets the network type for point-to-point. |

## Default Values

By default, Ethernet defaults to broadcast. PPP and Frame Relay default to point-to-point.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |
| Release 8.1 | ATM sub-interface was added. |

## Functional Notes

A point-to-point network will not elect designated routers.

## Usage Examples

The following example designates a broadcast network type:

(config)**#interface atm 1.1**
(config-atm 1.1)**#ip ospf network broadcast**

# ip pim sparse-mode

Use the **ip pim sparse-mode** command to enable protocol-independent multicast (PIM) sparse mode for this interface. Use the **no** form of this command to disable PIM sparse mode.

## Syntax Description

No subcommands.

## Default Values

By default, PIM sparse mode for this interface is disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Functional Notes

PIM Sparse Mode is a multicast routing protocol that makes use of the unicast forwarding table. It builds unidirectional shared trees rooted at a Rendezvous Point (RP) for a multicast group or a shortest-path tree rooted at a specific source for a multicast group.

## Usage Examples

The following example enables PIM sparse mode on the interface:

(config)**#interface atm 1.1**
(config-atm 1.1)**#ip pim sparse-mode**

# ip pim-sparse dr-priority <*value*>

Use the **ip pim-sparse dr-priority** command to specify the priority for the designated router (DR). This command modifies the routers priority in the DR election process. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the priority of this interface (to be used when determining the DR). Valid range is 1 to 4,294,967,295. |

## Default Values

By default, the priority of all PIM interfaces is 1.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Functional Notes

Interfaces advertise their configured priority values in the hello messages transmitted on the interface. Routers use the priority values to determine the appropriate DR. The router on the network segment with the highest priority is selected as the DR. If a hello message is received on the interface from a router on the network segment and it does not contain a priority, the entire network segment defaults to DR selection based on IP addresses instead of priority. In this instance, the DR is selected as the router on the network segment that has the highest IP address. AOS will always include a priority in all transmitted hello messages. If no priority is specifically designated by the user, the priority is set as the default of 1.

## Usage Examples

The following example specifies a priority of **100** on the ATM sub-interface 1.1:

(config)**#interface atm 1.1**
(config-atm 1.1)**#ip pim-sparse dr-priority 100**

# ip pim-sparse hello-timer <*value*>

Use the **ip pim-sparse hello-timer** command to specify protocol-independent multicast (PIM) sparse hello timer period. This is the time interval at which periodic hellos are sent out on all interfaces of a PIM-capable router. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the interval (in seconds) at which periodic hellos are sent out of the interface. Valid range is 10 to 3600 seconds. |

## Default Values

By default, the hellos are transmitted on PIM interfaces every 60 seconds.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Functional Notes

Hello messages are used to inform neighbors of a router's presence. Hello messages normally generate a small amount of traffic on an interface. Setting the **hello-timer** to a small interval increases the number of hellos sent (thus increasing the amount of traffic). Set the **hello-timer** to a reasonable value, taking into consideration the bandwidth available on the interface.

## Usage Examples

The following example specifies hellos be sent on the ATM sub-interface 1.1 every **3600** seconds:

(config)**#interface atm 1.1**
(config-atm 1.1)**#ip pim-sparse hello-timer 3600**

# ip pim-sparse nbr-timeout <*value*>

Use the **ip pim-sparse nbr-timeout** command to specify protocol-independent multicast (PIM) sparse neighbor timeout. This is the time interval after which a PIM-capable router will consider a neighbor not present. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the time interval in seconds after which a neighbor is considered not present. Valid range is 30 to 10,800 seconds. |

## Default Values

By default, the nbr-timeout is set to 105 seconds.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example sets the nbr-timeout to **300** seconds:

(config)**#interface atm 1.1**
(config-atm 1.1)**#ip pim-sparse nbr-timeout 300**

# ip pim-sparse override-interval <*value*>

Use the **ip pim-sparse override-interval** command to specify the protocol-independent multicast (PIM) sparse join/prune override interval. This delay interval is the period after a join/prune that another router on the LAN may override a join/prune. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the delay time in milliseconds. Valid range is 0 to 65,535 milliseconds. |

## Default Values

By default, the override interval is set to 2500 milliseconds.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example sets the override interval to **3000** milliseconds:

(config)**#interface atm 1.1**
(config-atm 1.1)**#ip pim-sparse override-interval 3000**

# ip pim-sparse propagation-delay <*value*>

Use the **ip pim-sparse propagation-delay** command to specify the expected propagation delay for join/prune messages. Set the propagation delay (in milliseconds) to estimate the amount of delay found in the local link. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the expected propagation delay in the local link in milliseconds. Valid range is 0 to 32,767 milliseconds. |

## Default Values

By default, the propagation delay is set to 500 milliseconds.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example sets the propagation delay to **300** milliseconds:

(config)**#interface atm 1.1**
(config-atm 1.1)**#ip pim-sparse propagation-delay 300**

# ip policy route-map *<name>*

Use the **ip policy route-map** command to assign a policy route-map to this interface. Use the **no** form of this command to remove the route-map policy.

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies the name of the policy route map to assign to this interface. |

## Default Values

By default, no policy route map is assigned to this interface.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example assigns the policy route map **policy1** to the ATM sub-interface 1.1:

(config)**#interface atm 1.1**
(config-atm 1.1)**#ip policy route-map policy1**

# ip proxy-arp

Use the **ip proxy-arp** to enable proxy Address Resolution Protocol (ARP) on the interface. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, proxy ARP is enabled.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 8.1 | ATM sub-interface was added. |

## Functional Notes

In general, the principle of proxy ARP allows a router to insert its IP address in the source IP address field of a packet (if the packet is from a host on one of its subnetworks). This allows hosts to reach devices on other subnetworks without implementing routing or specifying a default gateway.

If proxy ARP is enabled, the AOS will respond to all proxy ARP requests with its specified MAC address and forward packets accordingly.

Enabling proxy ARP on an interface may introduce unnecessary ARP traffic on the network.

## Usage Examples

The following enables proxy ARP on the ATM sub-interface 1.1:

(config)**#interface atm 1.1**
(config-atm 1.1)**#ip proxy-arp**

# ip rip receive version

Use the **ip rip receive version** command to configure the RIP version the unit accepts in all RIP packets received on the interface. Use the **no** form of this command to restore the default value. Variations of this command include:

**ip rip receive version 1**
**ip rip receive version 2**

## Syntax Description

| | |
|---|---|
| **1** | Accepts only received RIP version 1 packets on the interface. |
| **2** | Accepts only received RIP version 2 packets on the interface. |

## Default Values

By default, all interfaces implement RIP version 1 (the default value for the **version** command).

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 8.1 | ATM sub-interface was added. |

## Functional Notes

Use the **ip rip receive version** to specify a RIP version that will override the **version** (in the Router RIP) configuration.

The AOS only accepts one version (either 1 or 2) on a given interface.

## Usage Examples

The following example configures the ATM sub-interface 1.1 to accept only RIP version 2 packets:

(config)**#interface atm 1.1**
(config-atm 1.1)**#ip rip receive version 2**

# ip rip send version

Use the **ip rip send version** command to configure the RIP version the unit sends in all RIP packets transmitted on the interface. Use the **no** form of this command to restore the default value. Variations of this command include:

**ip rip send version 1**
**ip rip send version 2**

## Syntax Description

| | |
|---|---|
| **1** | Transmits only RIP version 1 packets on the interface |
| **2** | Transmits only RIP version 2 packets on the interface |

## Default Values

By default, all interfaces transmit RIP version 1 (the default value for the **version** command).

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 8.1 | ATM sub-interface was added. |

## Functional Notes

Use the **ip rip send version** to specify a RIP version that will override the **version** (in the Router RIP) configuration.

The AOS only transmits one version (either 1 or 2) on a given interface.

## Usage Examples

The following example configures the ATM sub-interface 1.1 to transmit only RIP version 2 packets:

(config)**#interface atm 1.1**
(config-atm 1.1)**#ip rip send version 2**

# ip rip summary-address *<ip address> <subnet mask>*

Use the **ip rip summary-address** command to manually summarize the routes Routing Information Protocol (RIP) will advertise and send out a specified interface. Use the **no** form of this command to disable this mode.

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies the summarized network IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.0**). |
| *<subnet mask>* | Specifies the subnet mask that corresponds to the range of IP addresses (network). Subnet masks can be expressed in dotted decimal notation (for example, **255.255.255.0**) or as a prefix length (for example, **/24**). |

## Default Values

By default, no manual summarization is applied by RIP.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Functional Notes

Unlike the automatic summarization on classful network boundaries, only specific network advertisements are made by RIP using the **ip rip summary-address** command. This command is only effective if RIP version 2 is configured.

## Usage Examples

The following example enables manual summarization on the specified IP address:

(config)**#interface atm 1.1**
(config-atm 1.1)**#ip rip summary-address 10.10.123.0 255.255.255.0**

# ip route-cache

Use the **ip route-cache** command to enable fast-cache switching on the interface. Use the **no** form of this command to disable fast-cache switching and return to process switching mode.

*Using Network Address Translation (NAT) or the AOS firewall capabilities on an interface requires process switching mode (using the **no ip route-cache** command).*

## Syntax Description

No subcommands.

## Default Values

By default, fast-cache switching is enabled on all Ethernet and virtual Frame Relay sub-interfaces. IP route-cache is enabled for all virtual PPP interfaces.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |
| Release 8.1 | ATM sub-interface was added. |

## Functional Notes

Fast switching allows an IP interface to provide optimum performance when processing IP traffic.

## Usage Examples

The following example enables fast switching on the ATM sub-interface 1.1:

(config)**#interface atm 1.1**
(config-atm 1.1)**#ip route-cache**

# ip unnumbered *<interface>*

Use the **ip unnumbered** command to use the IP address assigned to the specified interface for all IP processing on the active interface. Use the **no** form of this command to remove the unnumbered configuration.

## Syntax Description

| | |
|---|---|
| *<interface>* | Specifies the interface that contains the IP address to use as the source address for all packets transmitted on this interface. Specify an interface in the format *<interface type [slot/port \| slot/port.sub-interface id \| interface id \| interface id.sub-interface id]>*. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **show ip unnumbered ?** for a list of valid interfaces. |

## Default Values

By default, all interfaces are configured to use a specified IP address (using the **ip address** command).

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 8.1 | ATM sub-interface was added. |

## Functional Notes

If **ip unnumbered** is enabled on an interface, all IP traffic from the interface will use a source IP address taken from the specified interface. For example, specifying **ip unnumbered eth 0/1** while in the Frame Relay Sub-Interface Configuration mode configures the Frame Relay sub-interface to use the IP address assigned to the Ethernet interface for all IP processing. In addition, the AOS uses the specified interface information when sending route updates over the unnumbered interface.

## Usage Examples

The following example configures the ATM sub-interface 1.1 to use the IP address assigned to the Ethernet interface (**eth 0/1**):

(config)**#interface atm 1.1**
(config-atm 1.1)**#ip unnumbered eth 0/1**

# ip urlfilter *<name>*

Use the **ip urlfilter** command to apply a URL filter to the interface for all inbound or outbound traffic. Use the **no** form of this command to remove the URL filter from an interface. Variations of this command include:

**ip urlfilter** *<name>* **in**
**ip urlfilter** *<name>* **out**

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies the URL filter name to use on the interface. |
| **in** | Applies the filter to the inbound traffic. |
| **out** | Applies the filter to the outbound traffic. |

## Default Values

By default, there are no URL filters applied to any interfaces.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Functional Notes

The firewall must be enabled using the **ip firewall** command in order to use URL filters. The URL filter must be created by using the **ip urlfilter *<filtername>* http** command before applying it to the interface. See *ip urlfilter <name> http* on page 598 for more information on using this command.

## Usage Example

The following example performs URL filtering on all traffic entering through the ATM sub-interface 1.1 and matches the URL filter named **MyFilter**:

(config)**#interface atm 1.1**
(config-atm 1.1)**#ip urlfilter MyFilter in**

# max-reserved-bandwidth <*value*>

Use the **max-reserved-bandwidth** command to specify the percentage of interface bandwidth reserved for use in user-defined (priority or class-based) queues. The remainder of the interface bandwidth is reserved for system critical traffic and is not available to user-defined queues. Use the **no** form of this command to restore the default values.


*Reserving a portion of the interface bandwidth for system critical traffic is necessary for proper operation. Specifying the entire interface bandwidth for use in user-defined queues can cause undesirable operation.*

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the maximum percentage of bandwidth to reserve for QoS. This setting is configured as a percentage of the total interface speed. Range: 1 to 100 percent. |

## Default Values

By default, **max-reserved-bandwidth** is set to 75 percent, which reserves 25 percent of the interface bandwidth for system critical traffic.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example specifies **85** percent bandwidth on the ATM sub-interface to be available for use in user-defined queues:

(config)**#interface atm 1.1**
(config-atm 1.1)**#max-reserved-bandwidth 85**

# media-gateway ip

Use the **media-gateway ip** command to associate an IP address source to use for RTP traffic. When configuring VoIP, RTP traffic needs an IP address to be associated with it. However, some interfaces allow "dynamic" configuration of IP addresses, and thus, this value could change periodically. Use the **no** form of these commands to disable these functions. Variations of this command include:

**media-gateway ip loopback** <*ip address*>
**media-gateway ip primary**
**media-gateway ip secondary** <*ip address*>

## Syntax Description

| | |
|---|---|
| **loopback** <*ip address*> | Use an IP address statically defined to a loopback interface. Helpful when using a single IP address across multiple WAN interfaces for RTP traffic. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| **primary** | Use the IP address that is configured as primary on this interface. Applies to static, DHCP, or negotiated addresses. |
| **secondary** <*ip address*> | Use the statically defined secondary IP address of this interface to be used for RTP traffic. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

## Default Values

By default, **media-gateway ip** is disabled.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example configures the unit to use the primary IP address for RTP traffic:

(config)**#interface atm 1.1**
(config-atm 1.1)**#media-gateway ip primary**

# mtu <*size*>

Use the **mtu** command to configure the maximum transmit unit (MTU) size for the active interface. Use the **no** form of this command to return to the default value.

## Syntax Description

| | | |
|---|---|---|
| <*size*> | Configures the window size for transmitted packets. The valid ranges for the various interfaces are listed below: | |
| | ATM interfaces | 64 to 1520 |
| | Demand interfaces | 64 to 1520 |
| | Ethernet interfaces | 64 to 1500 |
| | FDL interfaces | 64 to 256 |
| | HDLC interfaces | 64 to 1520 |
| | Loopback interfaces | 64 to 1500 |
| | Tunnel interfaces | 64 to 18,190 |
| | Virtual Frame Relay sub-interfaces | 64 to 1520 |
| | Virtual PPP interfaces | 64 to 1500 |

## Default Values

| | | |
|---|---|---|
| <*size*> | The default values for the various interfaces are listed below: | |
| | ATM interfaces | 1500 |
| | Demand interfaces | 1500 |
| | Ethernet interfaces | 1500 |
| | FDL interfaces | 256 |
| | HDLC interfaces | 1500 |
| | Loopback interfaces | 1500 |
| | Tunnel interfaces | 1500 |
| | Virtual Frame Relay sub-interfaces | 1500 |
| | Virtual PPP interfaces | 1500 |

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 8.1 | ATM sub-interface was added. |

## Functional Notes

OSPF will not become adjacent on links where the MTU sizes do not match. If router A and router B are exchanging hello packets but their MTU sizes do not match, they will never reach adjacency. This is by design and required by the RFC.

## Usage Examples

The following example specifies an MTU of **1200** on the ATM sub-interface 1.1:

(config)**#interface atm 1.1**
(config-atm 1.1)**#mtu 1200**

# oam retry

Use the **oam retry** command to configure parameters related to Operation, Administration, and Maintenance (OAM) management for an ATM interface. Use the **no** form of this command to disable OAM management parameters. Variations of this command include:

**oam retry**
**oam retry** *<up value>*
**oam retry** *<up value> <down value>*
**oam retry** *<up value> <down value> <value>*

## Syntax Description>

| | |
|---|---|
| *<up value>* | Optional. Specifies the number of consecutive end-to-end F5 OAM loopback cell responses that must be received in order to change a PVC connection state to up. Range is 1 to 255. |
| *<down value>* | Optional. Specifies the number of consecutive end-to-end F5 OAM loopback cell responses that are not received in order to change a PVC state to down. Range is 1 to 255. |
| *<value>* | Optional. Specifies the frequency (in seconds) that end-to-end F5 OAM loopback cells are transmitted when a change in the up/down state of a PVC is being verified. Range is 1 to 600 seconds. |

## Default Values

By default, the up-count is set to 3, the down-count is set to 5, and the retry frequency is 1.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Usage Examples

The following example configures the OAM parameters with an up-count of **2**, down-count of **2**, and retry frequency of **10**:

(config)**#interface atm 1.1**
(config-atm 1.1)**#oam retry 2 2 10**

# oam-pvc managed

Use the **oam-pvc managed** command to enable end-to-end F5 Operation, Administration, and Maintenance (OAM) loopback cell generation and OAM management for an ATM interface. Use the **no** form of this command to disable generation of OAM loopback cells. Variations of this command include:

**oam-pvc managed**
**oam-pvc managed** <*value*>

## Syntax Description

| | |
|---|---|
| <*value*> | Optional. Specifies the time delay between transmitting OAM loopback cells. Range is 0 to 600 seconds. |

## Default Values

By default, the frequency is 1 second.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Usage Examples

The following example enables OAM loopback cell generation with a frequency of **5** seconds:

(config)**#interface atm 1.1**
(config-atm 1.1)**#oam-pvc managed 5**

# **pvc** <*VPI/VCI*>

Use the **pvc** command to select the ATM virtual link for this sub-interface. Use the **no** form of this command to remove the link.

## Syntax Description>

| | |
|---|---|
| <*VPI/VCI*> | Specifies the ATM network virtual path identifier (VPI) for this PVC and the ATM network virtual path identifier (VPI) for this PVC. The VPI value range is 0 to 255, and the VCI value range is 32 to 65,535. |

## Default Values

No default value is necessary for this command.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Usage Examples

The following example sets the VPI to **8** and the VCI to **35**:

(config)**#interface atm 1.1**
(config-atm 1.1)**#pvc 8/35**

# qos-policy out *<name>*

Use the **qos-policy out** command to apply a previously-configured QoS map to outgoing packets on an interface. Use the **no** form of this command to remove the map from the interface.

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies the name of a previously-created QoS map (refer to *qos map <name> <number>* on page 643 for more information). |

## Default Values

No default value is necessary for this command.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |
| Release 8.1 | ATM sub-interface was added. |

## Functional Notes

When a QoS policy is applied to an interface, it may be disabled if the interface bandwidth is not adequate to support the requested bandwidth on the map set. Once the bandwidth problem is resolved, the map will work again. The bandwidth will be rechecked on any of the following changes:

1. A priority or class-based entry is added to, deleted from, or changed in a QoS map set.
2. The interface bandwidth is changed by the **bandwidth** command on the interface.
3. A QoS policy is applied to an interface.
4. A cross-connect is created that includes an interface with a QoS policy.
5. The interface queuing method is changed to fair-queue to use weighted fair queuing.
6. The interface operational status changes.
7. The interface bandwidth changes for other reasons (e.g., when ADSL finishes training).

In order to prevent the map from being disabled in cases of temporary inadequate bandwidth (e.g., a single link goes down in a dual T1 multilink configuration where the map requests more than one T1's worth of bandwidth), the QoS map uses the maximum theoretical bandwidth on an interface, not the actual bandwidth at that time. This actually helps QoS keep higher priority class-based traffic working better than best-effort traffic when the bandwidth drops.

## Usage Examples

The following example applies the QoS map **VOICEMAP** to the ATM sub-interface 1.1:

(config)**#interface atm 1.1**
**(config-atm 1.1)#**qos-policy out VOICEMAP

# spanning-tree bpdufilter

Use the **spanning-tree bpdufilter** command to block BPDUs from being transmitted and received on this interface. To return to the default value, use the **no** form of this command. Variations of this command include:

**spanning-tree bpdufilter disable**
**spanning-tree bpdufilter enable**

## Syntax Description

| | |
|---|---|
| **disable** | Disables the BPDU filter. |
| **enable** | Enables the BPDU filter. |

## Default Values

By default, this command is set to **disable**.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |
| Release 8.1 | ATM sub-interface was added. |

## Functional Notes

The purpose of this command is to remove a port from participation in the spanning tree. This might be beneficial while debugging a network setup. It normally should not be used in a live network.

## Usage Examples

The following example enables the BPDU filter on the interface:

(config)**#interface atm 1.1**
(config-atm 1.1)**#spanning-tree bpdufilter enable**

# spanning-tree bpduguard

Use the **spanning-tree bpduguard** command to block BPDUs from being received on this interface. To return to the default value, use the **no** form of this command. Variations of this command include:

**spanning-tree bpduguard disable**
**spanning-tree bpduguard enable**

## Syntax Description

| | |
|---|---|
| **disable** | Disables the BPDU block. |
| **enable** | Enables the BPDU block. |

## Default Values

By default, this command is set to **disable**.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |
| Release 8.1 | ATM sub-interface was added. |

## Usage Examples

The following example enables the bpduguard on the interface:

(config)**#interface atm 1.1**
(config-atm 1.1)**#spanning-tree bpduguard enable**

# spanning-tree edgeport

Use the **spanning-tree edgeport** command to set this interface to be an edgeport. This command overrides the Global setting (refer to *spanning-tree edgeport default* on page 678). Use the **no** form of this command to return to the default value.

## Syntax Description

No subcommands.

## Default Values

By default, this command is set to disable.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |
| Release 8.1 | ATM sub-interface was added. |

## Usage Examples

The following example configures the interface to be an edgeport:

(config)**#interface atm 1.1**
(config-atm 1.1)**#spanning-tree edgeport**

An individual interface can be configured to not be considered an edgeport. For example:

(config)**#interface atm 1.1**
(config-atm 1.1)**#spanning-tree edgeport disable**

or

(config)**#interface atm 1.1**
(config-atm 1.1)**#no spanning-tree edgeport**

# spanning-tree link-type

Use the **spanning-tree link-type** command to configure the spanning-tree protocol link type for an interface. To return to the default value, use the **no** form of this command. Variations of this command include:

**spanning-tree link-type auto**
**spanning-tree link-type point-to-point**
**spanning-tree link-type shared**

## Syntax Description

| | |
|---|---|
| **auto** | Determines link type by the port's duplex settings. |
| **point-to-point** | Sets link type manually to point-to-point, regardless of duplex settings. |
| **shared** | Sets link type manually to shared, regardless of duplex settings. |

## Default Values

By default, a port is set to auto.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |
| Release 8.1 | ATM sub-interface was added. |

## Functional Notes

This command overrides the default link-type setting determined by the duplex of the individual port. By default a port configured for half-duplex is set to **shared** link type, and a port configured for full-duplex is set to **point-to-point** link type. Setting the link type manually overrides the default and forces the port to use the specified link type. Using the **link-type auto** command, restores the convention of determining link type based on duplex settings.

## Usage Examples

The following example forces the link-type to point-to-point, even if the port is configured to be half-duplex:

(config)**#bridge 1 protocol ieee**
(config)**#interface atm 1.1**
(config-atm 1.1)**#spanning-tree link-type point-to-point**

## Technology Review

Rapid transitions are possible in rapid spanning-tree protocol (RSTP) by taking advantage of point-to-point links (a port is connected to exactly one other bridge) and edge-port connections (a port is not connected to any additional bridges). Setting the link type to **auto** allows the spanning tree to automatically configure the link type based on the duplex of the link. Setting the link type to **point-to-point** allows a half-duplex link to act as if it were a point-to-point link.

# spanning-tree path-cost *<value>*

Use the **spanning tree path-cost** command to assign a cost to a bridge group that is used when computing the spanning-tree root path. To return to the default path-cost value, use the **no** form of this command.

## Syntax Description

| | |
|---|---|
| *<value>* | Assigns number to the bridge interface to be used as the path cost in spanning calculations. Valid range is 0 to 65,535. |

## Default Values

By default, the path-cost value is set to 19.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 8.1 | ATM sub-interface was added. |

## Functional Notes

The specified value is inversely proportional to the likelihood the bridge interface will be chosen as the root path. Set the path-cost value lower to increase the chance the interface will be the root. To obtain the most accurate spanning tree calculations, develop a system for determining path costs for links and apply it to all bridged interfaces.

## Usage Examples

The following example assigns a path cost of 100 for bridge group 17 on an ATM sub-interface:

(config)**#interface atm 1.1**
(config-atm 1.1)**#spanning-tree path-cost 100**

## Technology Review

Spanning-tree protocol provides a way to prevent loopback or parallel paths in bridged networks. Using the priority values and path costs assigned to each bridging interface, the spanning-tree protocol determines the root path and identifies whether to block or allow other paths.

# spanning-tree port-priority *<value>*

Use the **spanning-tree port-priority** command to select the priority level of a port associated with a bridge. To return to the default bridge-group priority value, use the **no** version of this command.

## Syntax Description

| | |
|---|---|
| *<value>* | Assigns a priority value for the bridge group; the lower the value, the higher the priority. Valid range is 0 to 255. |

## Default Values

By default, the bridge-group priority value is set to 128.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 8.1 | ATM sub-interface was added. |

## Functional Notes

The only time that this priority level is used is when two interfaces with a path to the root have equal cost. At that point, the level set in this command will determine which port the bridge will use. Set the priority value lower to increase the chance the interface will be used.

## Usage Examples

The following example sets the maximum priority on the ATM sub-interface labeled 1.1 in bridge group 17:

(config)**#interface atm 1.1**
(config-atm 1.1)**#spanning-tree priority 0**

# DEMAND INTERFACE CONFIGURATION COMMAND SET

To activate the Demand Interface Configuration mode, enter the **interface demand** command at the Global Configuration mode prompt. For example:

```
#configure terminal
(config)#interface demand 1
(config-demand 1)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:



All other commands for this command set are described in this section in alphabetical order.



# access-policy <*name*>

Use the **access-policy** command to assign a specified access policy to an interface. Access policies are applied to traffic entering an interface. Use the **no** form of this command to remove an access policy association. For more information on using access policies, refer to *ip policy-class <name>* on page 549.

*Configured access policies will only be active if the **ip firewall** command has been entered at the Global Configuration mode prompt to enable the AOS security features. All configuration parameters are valid, but no security data processing will be attempted unless the security features are enabled.*

## Syntax Description

| | |
|---|---|
| *<policyname>* | Identifies the configured access policy by alphanumeric descriptor (all access policy descriptors are case-sensitive). |

## Default Values

By default, there are no configured access policies associated with an interface.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include Ethernet sub-interfaces. |
| Release 6.1 | Command was expanded to include VLAN interfaces. |

## Functional Notes

To assign an access policy to an interface, enter the interface configuration mode for the desired interface and enter **access-policy** <*name*>.

## Usage Examples

The following example associates the access policy **Private** (to allow inbound traffic to the Web server) to the demand interface 1:

Enable the AOS security features:
(config)**#ip firewall**

Associate the access policy with the demand interface 1:

(config)**#interface demand 1**
(config-demand 1)**#access-policy Private**

# bandwidth *<value>*

Use the **bandwidth** command to provide the bandwidth value of an interface to the higher-level protocols. This value is used in cost calculations. Use the **no** form of this command to restore the default values.

## Syntax Description

| | |
|---|---|
| *<value>* | Specifies the bandwidth value in kbps. |

## Default Values

To view default values, use the **show interfaces** command.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |
| Release 11.1 | Command expanded to include the demand interface. |

## Functional Notes

The **bandwidth** command is an informational value that is communicated to the higher-level protocols to be used in cost calculations. This is a routing parameter only and does not affect the physical interface.

## Usage Examples

The following example sets the bandwidth of the demand interface to 10 Mbps:

```
(config)#interface demand 1
(config-demand 1)#bandwidth 10000
```

# called-number *<number>*

Use the **called-number** command to link calls to specific interfaces based on their dialed number identification service (DNIS) numbers. Multiple called numbers may be specified for an interface. Use the **no** form of this command to restore the default values.

## Syntax Description

| | |
|---|---|
| *<number>* | Identifies the called number to be linked to an interface. The DNIS number is limited to 20 digits. |

## Default Values

By default no called numbers are defined.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example links calls with a DNIS number of **2565558409** to the demand interface **1**:

(config)**#interface demand 1**
(config-demand 1)**#called-number 2565558409**

# caller-number <*number*>

Use the **caller-number** command to link calls to specific interfaces based on its caller ID (CLID) number. Multiple caller ID numbers may be specified, allowing the interface to accept calls from different remote resources. Use the **no** form of this command to restore the default values.

## Syntax Description

| | |
|---|---|
| <*number*> | Identifies the caller's number to be linked to an interface. The CLID number is limited to 20 digits. |

## Default Values

By default, no caller numbers are defined.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example links calls with a CLID number of **2565559911** to the demand interface **1**:

(config)**#interface demand 1**
(config-demand 1)**#caller-number 2565559911**

# connect-mode

Use the **connect-mode** command to configure the interface to only answer calls, only originate calls, or to both answer and originate calls. Use the **no** form of this command to restore the default values. Variations of this command include:

**connect-mode answer**
**connect-mode either**
**connect-mode originate**

## Syntax Description

| | |
|---|---|
| **answer** | Specifies the interface may be used to answer calls but not originate calls. |
| **either** | Specifies the interface may be used to answer and originate calls. |
| **originate** | Specifies the interface may be used to originate calls but not answer calls. |

## Default Values

By default the connect mode is set to both answer and originate calls.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example configures demand interface **1** to only answer calls:

(config)**#interface demand 1**
(config-demand 1)**#connect-mode answer**

# connect-order

Use the **connect-order** command to specify the starting point in the connection sequence for each sequence activation. The connection sequence is a circular list. Use the **no** form of this command to restore the default values. Variations of this command include:

**connect-order last-successful**
**connect-order round-robin**
**connect-order sequential**

## Syntax Description

| | |
|---|---|
| **last-successful** | Specifies the connect sequence be processed beginning with the last successful entry or the first entry if there are no previous connections. |
| **round-robin** | Specifies the connect sequence be processed beginning with the entry that follows the last successful entry or the first entry if there are no previous connections. |
| **sequential** | Specifies the connect sequence be processed from the beginning of the list. |

## Default Values

By default, connect sequences are processed sequentially.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example configures the connection sequence to begin with the last successful entry:

(config)**#interface demand 1**
(config-demand 1)**#connect-order last-successful**

# connect-sequence

Use the **connect-sequence** command to provide instructions to the interface on how to use the resource pool and telephone numbers to connect to demand destinations. Use the **no** form of this command to restore the default values.Variations of this command include the following:

**connect-sequence** <*value*> **dial-string** <*string*> **forced-analog**
**connect-sequence** <*value*> **dial-string** <*string*> **forced-analog busyout-threshold** <*value*>
**connect-sequence** <*value*> **dial-string** <*string*> **forced-isdn-56k**
**connect-sequence** <*value*> **dial-string** <*string*> **forced-isdn-56k busyout-threshold** <*value*>
**connect-sequence** <*value*> **dial-string** <*string*> **forced-isdn-64k**
**connect-sequence** <*value*> **dial-string** <*string*> **forced-isdn-64k busyout-threshold** <*value*>
**connect-sequence** <*value*> **dial-string** <*string*> **isdn-56k**
**connect-sequence** <*value*> **dial-string** <*string*> **isdn-56k busyout-threshold** <*value*>
**connect-sequence** <*value*> **dial-string** <*string*> i**sdn-64k**
**connect-sequence** <*value*> **dial-string** <*string*> **isdn-64k busyout-threshold** <*value*>

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the sequence number for this connection specification entry. Range is 1 to 65,535. |
| **dial-string** <*string*> | Specifies the telephone number to dial when using this connection. The dial string is limited to 20 digits. |
| **forced-analog** | Specifies that only analog resources may be used. |
| **forced-isdn-56k** | Specifies that only ISDN resources may be used. Call is placed using ISDN 56k. |
| **forced-isdn-64k** | Specifies that only ISDN resources may be used. Call is placed using ISDN 64k. |
| **isdn-56k** | Specifies any dial resource may be used if ISDN 56k call-type is used. |
| **isdn-64k** | Specifies any dial resource may be used if ISDN 64k call-type is used. |
| **busy-threshold** <*value*> | Optional. Specifies the maximum number of connect sequence cycles during a activation attempt that must fail before it is skipped until the next activation attempt. |

## Default Values

By default any dial resource may be used.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example instructs demand interface **1** to place the call using ISDN 64k:

(config)**#interface demand 1**
(config-demand 1)**#connect-sequence 65 dial-string 2565559911 forced-isdn-64k**

# connect-sequence attempts *<value>*

Use the **connect-sequence attempts** command to limit the number of times the connect sequence will cycle when its entries are unable to establish a connection. When the maximum number of attempts are exhausted, the interface will go into recovery mode. Refer to *connect-sequence interface-recovery* on page 1099 for more information. Use the **no** form of this command to restore the default values.

## Syntax Description

| | |
|---|---|
| *<value>* | Specifies the number of times the connect sequence will cycle through its entries if it is unable to make a connection. Range is 0 to 65,535. |

## Default Values

By default the connect-sequence attempts are unlimited.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example instructs demand interface **1** to attempt its connection sequence **500** times:

(config)**#interface demand 1**
(config-demand 1)**#connect-sequence attempts 500**

# connect-sequence interface-recovery

Use the **connect-sequence interface-recovery** command to allow the interface to go down in the event that the **connect-sequence attempts** value is exhausted. Refer to *connect-sequence attempts <value>* on page 1098 for more information. Use the **no** form of this command to restore the default values.Variations of this command include:

**connect-sequence interface-recovery**
**connect-sequence interface-recovery retry-interval** <*value*>
**connect-sequence interface-recovery retry-interval** <*value*> **max-retries** <*number*>

## Syntax Description

| | |
|---|---|
| **retry-interval** <*value*> | Optional. Specifies the number of seconds the interface will wait between connect sequence cycles during recovery attempts. |
| **max-retries** <*number*> | Optional. Specifies the maximum number of times the connect sequence will cycle in an attempt to bring the interface back up. When in interface recovery mode, this value overrides the **connect-sequence attempts** value. |

## Default Values

By default, the **connect-sequence interface-recovery retry-interval** is set to 120 seconds and **max-retries** are unlimited.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example configures demand interface **1** to wait **60** seconds between retry attempts with a maximum number of **500** retries:

(config)**#interface demand 1**
(config-demand 1)**#connect-sequence interface-recovery retry-interval 60 max-retries 500**

# crypto map *<name>*

Use the **crypto map** command to associate crypto maps with the interface. Use the **no** form of this command to remove a crypto map from an interface.

> NOTE *When you apply a map to an interface, you are applying all crypto maps with the given map name. This allows you to apply multiple crypto maps if you have created maps which share the same name but have different map index numbers.*

> NOTE *For VPN configuration example scripts, refer to the technical support note **Configuring VPN** located on the ADTRAN OS System Documentation CD provided with your unit.*

## Syntax Description

| | |
|---|---|
| *<name>* | Assigns a crypto map name to the interface. |

## Default Values

By default, no crypto maps are assigned to an interface.

## Command History

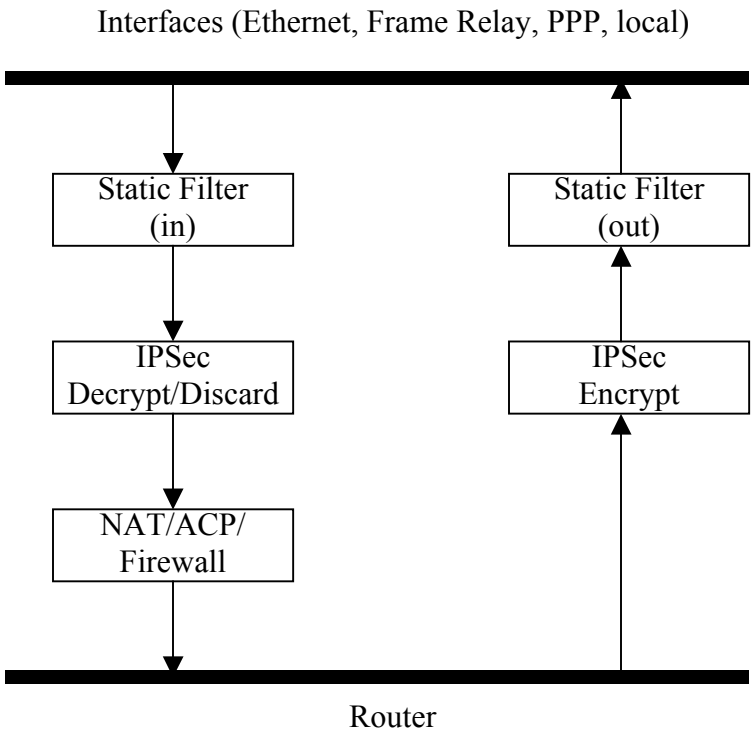
| | |
|---|---|
| Release 4.1 | Command was introduced. |
| Release 11.1 | Command expanded to include the demand interface. |

## Functional Notes

When configuring a system to use both the stateful inspection firewall and IKE negotiation for VPN, keep the following notes in mind.

When defining the policy class and associated access-control lists (ACLs) that describe the behavior of the firewall, do not forget to include the traffic coming into the system over a VPN tunnel terminated by the system. The firewall should be set up with respect to the unencrypted traffic that is destined to be sent or received over the VPN tunnel. The following diagram represents typical AOS data-flow logic.




As shown in the diagram above, data coming into the product is first processed by the static filter associated with the interface on which the data is received. This access group is a true static filter and is available for use regardless of whether the firewall is enabled or disabled. Next (if the data is encrypted) it is sent to the IPSec engine for decryption. The decrypted data is then processed by the stateful inspection firewall. Therefore, given a terminating VPN tunnel, only unencrypted data is processed by the firewall.

The ACLs for a crypto map on an interface work in reverse logic to the ACLs for a policy class on an interface. When specifying the ACLs for a crypto map, the source information is the private local-side, unencrypted source of the data. The destination information will be the far end, unencrypted destination of the data. However, ACLs for a policy class work in reverse. The source information for the ACL in a policy class is the far end. The destination information is the local side.

## Usage Examples

The following example applies all crypto maps with the name **MyMap** to the demand interface:

(config)**#interface demand 1**
(config-demand 1)**#crypto map MyMap**

# demand-hold-queue <*number*> timeout <*value*>

Use the **demand-hold-queue timeout** command to set the number and length of time interesting packets will be held while a connection is being made. Use the **no** form of this command to restore the default values.

## Syntax Description

| | |
|---|---|
| <*number*> | Specifies the number of packets that may be stored in the hold queue. Range is 0 to 100. |
| <*value*> | Specifies the number of seconds a packet may remain in the hold queue. Range is 0 to 255 seconds. |

## Default Values

By default, the hold queue is disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example configures demand interface **1** to hold **50** packets in the queue for up to **120** seconds:

(config)**#interface demand 1**
(config-demand 1)**#demand-hold-queue 50 timeout 120**

# dynamic-dns

Use the **dynamic-dns** command to configure Dynamic DNS service provided by Dynamic Network Services, Inc. (www.dyndns.org). Use the **no** versions of these commands to disable these features. Variations of this command include:

**dynamic-dns custom** <*hostname*> <*value*>
**dynamic-dns dyndns** <*hostname*> <*username*> <*password*>
**dynamic-dns dyndns-custom** <*hostname*> <*username*> <*password*>
**dynamic-dns dyndns-static** <*hostname*> <*username*> <*password*>

## Syntax Description

| | |
|---|---|
| <*hostname*> | Specifies the host name for the server that updates the Dynamic Domain Name Server (DNS). |
| <*value*> | Specifies the intervals in minutes to update the server with information (updates also occur when the interface's IP address changes regardless of the update intervals).<br>Refer to *Functional Notes* below for additional argument descriptions. |

## Default Values

No default is necessary for this command.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |
| Release 12.1 | Command was expanded. |

## Functional Notes

**custom** - Constanttime.com's Custom Dynamic DNS$^{SM}$ service allows you complete access and management control over your domain name regardless of where you purchased/registered it. This allows you to manage IP address mappings (A records), domain aliases (CNAME records) and mail servers (MX records).

**dyndns** - The Dynamic DNS$^{SM}$ offered by Dynamic Network Services Inc., (DynDNS.org) allows you to alias a dynamic IP address to a static host name in various domains. This allows your unit to be more easily accessed from various locations on the Internet. This service is provided for up to five host names.

**dyndns-custom -** DynDNS.org's Custom DNS$^{SM}$ service provides a full DNS solution, giving you complete control over an entire domain name. A web-based interface provides two levels of control over your domain, catering to average or advanced users. Five globally redundant DNS servers ensure that your domain will always resolve.

A choice of two interfaces is available. The basic interface is designed for most users. It comes preconfigured for the most common configuration and allows for easy creation of most common record types. The advanced interface is designed for system administrators with a solid DNS background, and provides layout and functionality similar to a BIND zone file allowing for the creation of nearly any record type.

Custom DNS$^{SM}$ can be used with both static and dynamic IPs and has the same automatic update capability through Custom DNS-aware clients as Dynamic DNS.

**dyndns-static -** The Static DNS service is similar to DynDNS.org's Dynamic DNS$^{SM}$ service in that it allows a host name such as yourname.dyndns.org to point to your IP address. Unlike a Dynamic DNS host, a Static DNS host does not expire after 35 days without updates, but updates take longer to propagate though the DNS system. This service is provided for up to five host names.

If your IP address does not change often or at all but you still want an easy name to remember it by (without having to purchase your own domain name), Static DNS service is ideal for you.

If you would like to use your own domain name (such as yourname.com), you need Custom DNS service which also provides full dynamic and static IP address support.

**Usage Examples**

The following example sets the dynamic-dns to **dyndns-custom** with host name **host**, user name **user**, and password **pass**:

(config)**#interface demand 1**
(config-demand 1)**#dynamic-dns dyndns-custom host user pass**

# fair-queue

Use the **fair-queue** command to enable weighted fair queuing (WFQ) on an interface. Use the **no** form of this command to disable WFQ and enable FIFO queueing for an interface. WFQ is enabled by default for WAN interfaces. Variations of this command include:

**fair-queue**
**fair-queue** *<threshold>*

## Syntax Description

| | |
|---|---|
| *<threshold>* | Optional. Specifies the maximum number of packets that can be present in each conversation sub-queue. Packets received for a conversation after this limit is reached are discarded. Range: 16 to 512 packets. |

## Default Values

By default, WFQ is enabled with a threshold of 64 packets.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |
| Release 11.1 | Command expanded to include the demand interface. |

## Usage Examples

The following example enables WFQ on the interface with a threshold set at 100 packets:

(config)**#interface demand 1**
(config-demand 1)**#fair-queue 100**

# fast-idle *<value>*

Use the **fast-idle** command to set the amount of time the demand interface connection will remain active in the absence of interesting traffic when there is contention for the demand resources being used by this interface. Use the **no** form of this command to restore the default values.

## Syntax Description

| | |
|---|---|
| *<value>* | Specifies the number of seconds the interface will remain up in the absence of interesting traffic. Range is 1 to 2,147,483 seconds. |

## Default Values

By default, **fast-idle** is set to 120 seconds.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example sets fast idle to 1,073,752 seconds:

(config)**#interface demand 1**
(config-demand 1)**#fast-idle 1073752**

# hold-queue <*value*> out

Use the **hold-queue out** command to change the overall size of an interface's WAN output queue. Use the **no** form of this command to return to the default settings.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the total number of packets the output queue can contain before packets are dropped. Range is 16 to 1000 packets. |

## Default Values

The default queue size for WFQ is 400. The default queue size for PPP FIFO and Frame Relay round-robin is 200.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |
| Release 11.1 | Command expanded to include the demand interface. |

## Usage Examples

The following example sets the overall output queue size to **700**:

(config)**#interface demand 1**
(config-demand 1)**#hold-queue 700 out**

# idle-timeout *<value>*

Use the **idle-timeout** command to set the amount of time the interface link/bundle will remain up in the absence of interesting traffic. Interesting traffic and direction logic are set using the **match-interesting** commands. Refer to *match-interesting* on page 1138 for more information. Use the **no** form of this command to restore the default values.

## Syntax Description

| | |
|---|---|
| *<value>* | Specifies the number of seconds the interface will remain up in the absence of interesting traffic. Range is 1 to 2,147,483 seconds. |

## Default Values

By default, **idle-timeout** is set to 120 seconds.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example configures demand interface **1** to time out after **360** seconds:

(config)**#interface demand 1**
(config-demand 1)**#idle-timeout 360**

# ip access-group *<name>*

Use the **ip access-group** command to create an access control list to be used for packets transmitted on or received from the specified interface. Use the **no** form of this command to disable this type of control. Variations of this command include:

**ip access-group** <*name*> **in**
**ip access-group** <*name*> **out**

## Syntax Description

| | |
|---|---|
| *<name>* | Indicates the assigned IP access control list name. |
| **in** | Enables access control on packets received on the specified interface. |
| **out** | Enables access control on packets transmitted on the specified interface. |

## Default Values

By default, these commands are disabled.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |
| Release 11.1 | Command expanded to include the demand interface. |

## Functional Notes

When this command is enabled, the IP destination address of each packet must be validated before being passed through. If the packet is not acceptable per these settings, it is dropped.

## Usage Examples

The following example sets up the router to only allow Telnet traffic into the demand interface:

(config)**#ip access-list extended TelnetOnly**
(config-ext-nacl)**#permit tcp any any eq telnet**
(config-ext-nacl)**#interface demand 1**
(config-demand 1)**#ip access-group TelnetOnly in**

# ip address negotiated

Use the **ip address negotiated** command to allow the interface to negotiate (i.e., be assigned) an IP address from the far end PPP connection. Use the **no** form of this command to disable the negotiation for an IP address. Variations of this command include:

**ip address negotiated**
**ip address negotiated no-default**

## Syntax Description

| | |
|---|---|
| **no-default** | Optional. Prevents the insertion of a default route. Some systems already have a default route configured and need a static route to the PPP interface to function correctly. |

## Default Values

By default, the interface is assigned an address with the **ip address** <*ip address*> <*subnet mask*> command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |
| Release 11.1 | Command expanded to include the demand interface. |

## Usage Examples

The following example enables the demand interface to negotiate an IP address from the far end connection:

(config)**#interface demand 1**
(config-demand 1)**#ip address negotiated**

The following example enables the demand interface to negotiate an IP address from the far end connection without inserting a default route:

(config)**#interface demand 1**
(config-demand 1)**#ip address negotiated no-default**

# ip address *<ip address> <subnet mask>*

Use the **ip address** command to define an IP address on the specified interface. Use the optional keyword **secondary** to define a secondary IP address. Use the **no** form of this command to remove a configured IP address. Variations of this command include:

**ip address** *<ip address> <subnet mask>*
**ip address** *<ip address> <subnet mask>* **secondary**

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies a valid IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| *<subnet mask>* | Specifies the subnet mask that corresponds to a range of IP addresses (network) or a specific host. Subnet masks can be expressed in dotted decimal notation (for example, **255.255.255.0**) or as a prefix length (for example, **/24**). |
| **secondary** | Optional. Configures a secondary IP address for the specified interface. |

## Default Values

By default, there are no assigned IP addresses.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 11.1 | Command expanded to include the demand interface. |

## Functional Notes

Use secondary IP addresses to allow dual subnets on a single interface (when you need more IP addresses than the primary subnet can provide). When using secondary IP addresses, avoid routing loops by verifying that all devices on the network segment are configured with secondary IP addresses on the secondary subnet.

## Usage Examples

The following example configures a secondary IP address of **192.22.72.101/30**:

(config)**#interface demand 1**
(config-demand 1)**#ip address 192.22.72.101 255.255.255.252 secondary**

# ip ffe

Use the **ip ffe** command to enable the FastFlow Engine (FFE) on this interface with the default number of entries. Use the **no** form of these commands to disable the FFE. Variations of this command include:

**ip ffe max-entries** <*entries*>

*Issuing this command will cause all FFE entries on this interface to be cleared.*

## Syntax Description

| | |
|---|---|
| **max-entries** <*entries*> | Specifies the maximum number of entries stored in the flow table. Valid range is from 1 to 8192. |

## Default Values

By default, the FFE is disabled. The default number of **max-entries** is 4096.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Functional Notes

The FFE can be used to help reduce routing overhead and thus reduce overall routing times. Routing times are reduced by the creation of a flow table on the ingress interface. The maximum number of entries that can be stored in the flow table at any one time may be specified by using the **max-entries** subcommand. When features such as firewall, VPN, and static filters are enabled, routing time reduction may not be realized.

## Usage Examples

The following example enables the FFE and sets the maximum number of entries in the flow table to 50:

(config)**#interface demand 1**
(config-demand 1)**#ip ffe max-entries 50**

## Technology Review

The FastFlow system goal is to increase IP packet throughput by moving as much of the packet processing into the engine as possible. Packets are classified into flows based upon the IP protocol (TCP, UDP, ICMP, etc.), the source and destination IP addresses, IP TOS, and the protocol-specific information such as the source and destination port numbers. Flows are defined as the unidirectional representation of a conversation between two IP hosts. Each ingress interface keeps its own flow table, a collection of flow entries.

The first packet in a flow that is forwarded through the unit will build a flow entry. When a flow entry is looked up but no entry is found, a FastFlowBuilder object is allocated and attached to the packet. As the packet passes through the various processing layers, each subsystem will add processing to the FastFlowBuilder. When packet is about to be forwarded out of the egress interface, the FastFlowBuilder will be finalized. That is, the flow entry being built will be checked for completeness and committed to the flow table on the ingress interface. Subsequent flow matches can then bypass the normal processing layers.

# ip helper-address *<ip address>*

Use the **ip helper-address** command to configure the AOS to forward User Datagram Protocol (UDP) broadcast packets received on the interface. Use the **no** form of this command to disable forwarding packets.

> NOTE *The **ip helper** command must be used in conjunction with the **ip forward-protocol** command to configure the AOS to forward UDP broadcast packets. Refer to ip forward-protocol udp <value> on page 532 for more information.*

## Syntax Description

| | |
|---|---|
| *<address>* | Specifies the destination IP address for the forwarded UDP packets. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

## Default Values

By default, broadcast UDP packets are not forwarded.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 11.1 | Command expanded to include the demand interface. |

## Functional Notes

When used in conjunction with the **ip forward-protocol** command, the **ip helper-address** feature allows you to customize which broadcast packets are forwarded.

To implement the helper address feature, assign a helper address(es) (specifying the device that needs to receive the broadcast traffic) to the interface closest to the host that transmits the broadcast packets. When broadcast packets (of the specified type forwarded using the **ip forward-protocol** command) are received on the interface, they will be forwarded to the device that needs the information.

Only packets meeting the following criteria are considered eligible by the **ip helper-address** feature:

1. The packet IP protocol is User Datagram Protocol (UDP).
2. Any UDP port specified using the **ip forward-protocol** command.
3. The media access control (MAC) address of the frame is an all-ones broadcast address (ffff.ffff.ffff).
4. The destination IP address is broadcast defined by all ones (255.255.255.255) or a subnet broadcast (for example, 192.33.4.251 for the 192.33.4.248/30 subnet).

## Usage Examples

The following example forwards all DNS broadcast traffic to the DNS server with IP address **192.33.5.99**:

(config)**#ip forward-protocol udp domain**
(config)**#interface demand 1**
(config-demand 1)**#ip helper-address 192.33.5.99**

# ip igmp

Use the **ip igmp** command to configure multicasting-related functions for the interface. Variations of this command include:

**ip igmp immediate-leave**
**ip igmp last-member-query-interval** <*milliseconds*>
**ip igmp querier-timeout** <*seconds*>
**ip igmp query-interval** <*seconds*>
**ip igmp query-max-response-time** <*seconds*>
**ip igmp static-group** <*address*>
**ip igmp version [1 | 2]**

## Syntax Description

| | |
|---|---|
| **immediate-leave** | Specifies that if only one host (or IGMP snooping switch) is connected to the interface, when a leave is received, multicast of that group is immediately terminated as opposed to sending a group query and timing out the group if no device responds. Works in conjunction with **ip igmp last-member-query-interval**. Applies to all groups when configured. Use the **no** form of this command to disable the **immediate-leave** feature. |
| **last-member-query-interval** <*milliseconds*> | Controls the timeout (in milliseconds) used to detect whether any group receivers remain on an interface after a receiver leaves a group. If a receiver sends a leave-group message (IGMP Version 2), the router sends a group-specific query on that interface. After twice the time specified by this command plus as much as one second longer, if no receiver responds, the router removes that interface from the group and stops sending that group's multicast packets to the interface. Range is 100 to 65,535 ms. Use the **no** form of this command to return to the default setting. |
| **querier-timeout** <*seconds*> | Specifies the interval (in seconds) that the router waits after the current querier's last query before it takes over as querier (IGMP V2). Range is 60 to 300 seconds. Use the **no** form of this command to return to the default setting. |
| **query-interval** <*seconds*> | Specifies the interval (in seconds) at which IGMP queries are sent on an interface. Host query messages are addressed to the all-hosts multicast group with an IP TTL of 1. The router uses queries to detect whether multicast group members are on the interface and to select an IGMP designated router for the attached segment (if more than one multicast router exists). Only the designated router for the segment sends queries. For IGMP V2, the designated router is the router with the lowest IP address on the segment. Range is 0 to 65,535 seconds. Use the **no** form of this command to return to the default setting. |
| **query-max-response-time** <*seconds*> | Specifies the maximum response time (in seconds) advertised by this interface in queries when using IGMP V2. Hosts are allowed a random time within this period to respond, reducing response bursts. Use the **no** form of this command to return to the default setting. |

## Syntax Description

| | |
|---|---|
| **static-group** <*address*> | Configures the router's interface to be a statically-connected member of the specified group. Packets received on the correct RPF interface are forwarded to this interface regardless of whether any receivers have joined the specified group using IGMP. Use the **no** form of this command to remove a configured static group. |
| **version [1 \| 2]** | Sets the interface's IGMP version. Use the **no** form of this command to return to the default setting. |

## Default Values

| | |
|---|---|
| **ip igmp immediate-leave** | No default |
| **ip igmp last-member-query-interval** | 1000 milliseconds |
| **ip igmp querier-timeout** | 2x the **query-interval** value |
| **ip igmp query-interval** | 60 seconds |
| **ip igmp query-max-response-time** | 10 seconds |
| **ip igmp static-group** | No default |
| **ip igmp version** | Version 1 |

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |
| Release 11.1 | Command expanded to include the demand interface. |

## Usage Examples

The following example sets the query message interval on the interface to 200 milliseconds:

(config)**#interface demand 1**
(config-demand 1)**#ip igmp last-member-query-interval 200**

# ip mcast-stub downstream

Use the **ip mcast-stub downstream** command to enable multicast forwarding and IGMP (router mode) on an interface and place it in multicast stub downstream mode. Use the **no** form of this command to disable.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |
| Release 11.1 | Command expanded to include the demand interface. |

## Functional Notes

This command is used in IP multicast stub applications in conjunction with the **ip mcast-stub helper-address** and **ip mcast-stub upstream** commands. Downstream interfaces connect to segments with multicast hosts. Multiple interfaces may be configured in downstream mode; however, interfaces connecting to the multicast network (upstream) should not be configured in downstream mode. Interfaces configured as downstream should have the lowest IP address of all IGMP-capable routers on the connected segment in order to be selected as the designated router and ensure proper forwarding. Refer to *ip mcast-stub helper-address <ip address>* on page 546 and *ip mcast-stub upstream* on page 1121 for more information.

## Usage Examples

The following example enables multicast forwarding and IGMP on the interface:

(config)**#interface demand 1**
(config-demand 1)**#ip mcast-stub downstream**

# ip mcast-stub fixed

Use the **ip mcast-stub fixed** command to allow forwarding of multicast traffic on a selected interface after enabling multicast routing. Use the **no** form of this command to disable this mode.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Functional Notes

Multicast routing must be enabled prior to setting **ip mcast-stub fixed** on the selected interface. Also, use the **ip igmp static-group** <*A.B.C.D*> command to receive multicast traffic without host-initiated Internet Group Management Protocol (IGMP) activity on the selected interface. Otherwise, all host-initiated IGMP transactions will enter multicast routes on the router's interface involved with IGMP activities.

### Usage Examples

The following example enables multicast traffic forwarding and IGMP on the interface:

(config)**#interface demand 1**
(config-demand 1)**#ip mcast-stub fixed**

# ip mcast-stub helper-enable

Use the **ip mcast-stub helper-enable** command to assign the **ip mcast-stub helper-address** as the IGMP proxy. Use the **no** form of this command to disable.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |
| Release 11.1 | Command was expanded to include the demand interface. |

## Functional Notes

This command is used in IP multicast stub applications in conjunction with the **ip mcast-stub helper-address**, **ip mcast-stub upstream**, and **ip mcast-stub downstream** commands. When enabled, the interface becomes a helper forwarding interface. The IGMP host function is dynamically enabled and the interface becomes the active upstream interface, enabling the unit to perform as an IGMP proxy. Refer to *ip mcast-stub helper-address <ip address>* on page 546, *ip mcast-stub downstream* on page 1118, and *ip mcast-stub upstream* on page 1121 for more information.

## Usage Examples

The following example sets the helper address as the IGMP proxy:

(config)**#interface demand 1**
(config-demand 1)**#ip mcast-stub helper-enable**

# ip mcast-stub upstream

Use the **ip mcast-stub upstream** command to enable multicast forwarding on an interface and place it in multicast stub upstream mode. Use the **no** form of this command to disable.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |
| Release 11.1 | Command expanded to include the demand interface. |

## Functional Notes

This command is used in IP multicast stub applications in conjunction with the **ip mcast-stub helper-address** and **ip mcast-stub downstream** commands. When enabled, the interface becomes a candidate to be a helper forwarding interface. If chosen as the best path toward the helper address by the router's unicast route table, the IGMP host function is dynamically enabled and the interface becomes the active upstream interface, enabling the router to perform as an IGMP proxy. Though multiple interfaces may be candidates, no more than one interface will actively serve as the helper forwarding interface. Refer to *ip mcast-stub helper-address <ip address>* on page 546 and *ip mcast-stub downstream* on page 1118 for more information.

## Usage Examples

The following example enables multicast forwarding on the interface:

(config)**#interface demand 1**
(config-demand 1)**#ip mcast-stub upstream**

# ip ospf

Use the **ip ospf** command to customize OSPF settings (if needed). Use the **no** form of these commands to return to the default settings. Variations of this command include:

**ip ospf authentication-key** <*password*>
**ip ospf cost** <*value*>
**ip ospf dead-interval** <*seconds*>
**ip ospf hello-interval** <*seconds*>
**ip ospf message-digest-key [1 | 2] md5** <*key*>
**ip ospf priority** <*value*>
**ip ospf retransmit-interval** <*seconds*>
**ip ospf transmit-delay** <*seconds*>

## Syntax Description

| | |
|---|---|
| **authentication-key** <*password*> | Assigns a simple-text authentication password to be used by other routers using the OSPF simple password authentication. |
| **cost** <*value*> | Specifies the OSPF cost of sending a packet on the interface. This value overrides any computed cost value. Range is 1 to 65,535. |
| **dead-interval** <*seconds*> | Sets the maximum interval (in seconds) allowed between hello packets. If the maximum is exceeded, neighboring devices will determine that the device is down. Range is 0 to 32,767 seconds. |
| **hello-interval** <*seconds*> | Specifies the interval (in seconds) between hello packets sent on the interface. Range is 0 to 32,767 seconds. |
| **message-digest-key [1 \| 2] md5** <*key*> | Configures OSPF Message Digest 5 (MD5) authentication (16-byte max) keys. |
| **priority** <*value*> | Sets the OSPF priority. The value set in this field helps determine the designated router for this network. Range is 0 to 255. |
| **retransmit-interval** <*seconds*> | Specifies the interval (in seconds) between link-state advertisements (LSAs). Range is 0 to 32,767 seconds. |
| **transmit-delay** <*seconds*> | Sets the estimated time (in seconds) required to send an LSA on the interface. Range is 0 to 32,767 seconds. |

## Default Values

| | |
|---|---|
| **dead-interval** <*seconds*> | 40 seconds |
| **hello-interval** <*seconds*> | 10 seconds: Ethernet, point-to-point, Frame Relay, and PPP |
| r**etransmit-interval** <*seconds*> | 5 seconds |
| **transmit-delay** <*seconds*> | 1 second |

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |
| Release 11.1 | Command expanded to include the demand interface. |

## Usage Example

The following example sets the maximum number of seconds allowed between hello packets to 25,000:

(config)**#interface demand 1**
(config-demand 1)**#ip ospf dead-interval 25000**

# ip ospf authentication

Use the **ip ospf authentication** command to authenticate an interface that is performing OSPF authentication. Use the **no** form of this command to return to the default setting. Variations of this command include:

**ip ospf authentication**
**ip ospf authentication message-digest**
**ip ospf authentication null**

## Syntax Description

| | |
|---|---|
| **message-digest** | Optional. Selects message-digest authentication type. |
| **null** | Optional. Specifies that no authentication be used. |

## Default Values

By default, **ip ospf authentication** is set to null (meaning no authentication is used).

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |
| Release 11.1 | Command expanded to include the demand interface. |

## Usage Examples

The following example specifies that no authentication will be used on the demand interface:

(config)**#interface demand 1**
(config-demand 1)**#ip ospf authentication null**

# ip ospf network

Use the **ip ospf network** command to specify the type of network on this interface. Use the **no** form of this command to return to the default setting. Variations of this command include:

**ip ospf network broadcast**
**ip ospf network point-to-point**

## Syntax Description

| | |
|---|---|
| **broadcast** | Sets the network type for broadcast. |
| **point-to-point** | Sets the network type for point-to-point. |

## Default Values

By default, Ethernet defaults to broadcast. PPP and Frame Relay default to point-to-point.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |
| Release 11.1 | Command expanded to include the demand interface. |

## Functional Notes

A point-to-point network will not elect designated routers.

## Usage Examples

The following example designates a broadcast network type:

(config)**#interface demand 1**
(config-demand 1)**#ip ospf network broadcast**

# ip policy route-map *<name>*

Use the **ip policy route-map** command to associate a route map with a network interface source. Use the **no** form of this command to disable this feature.

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies the route map to associate with this interface. |

## Default Values

By default, policy-based routing is disabled for all interfaces.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example associates the route map named **MyMap** with demand interface **1**:

(config)**#interface demand 1**
(config-demand 1)**#ip policy route-map MyMap**

# ip proxy-arp

Use the **ip proxy-arp** to enable proxy Address Resolution Protocol (ARP) on the interface. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, proxy ARP is enabled.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 11.1 | Command expanded to include the demand interface. |

## Functional Notes

In general, the principle of proxy ARP allows a router to insert its IP address in the source IP address field of a packet (if the packet is from a host on one of its subnetworks). This allows hosts to reach devices on other subnetworks without implementing routing or specifying a default gateway.

If proxy ARP is enabled, the AOS will respond to all proxy ARP requests with its specified MAC address and forward packets accordingly.

Enabling proxy ARP on an interface may introduce unnecessary ARP traffic on the network.

## Usage Examples

The following example enables proxy ARP on the virtual demand interface:

(config)**#interface demand 1**
(config-demand 1)**#ip proxy-arp**

# ip rip receive version

Use the **ip rip receive version** command to configure the RIP version the unit accepts in all RIP packets received on the interface. Use the **no** form of this command to restore the default value. Variations of this command include:

**ip rip receive version 1**
**ip rip receive version 2**

## Syntax Description

| | |
|---|---|
| **1** | Accepts only received RIP version 1 packets on the interface. |
| **2** | Accepts only received RIP version 2 packets on the interface. |

## Default Values

By default, all interfaces implement RIP version 1 (the default value for the **version** command).

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 11.1 | Command expanded to include the demand interface. |

## Functional Notes

Use the **ip rip receive version** to specify a RIP version that overrides the **version** (in the Router RIP) configuration.

The AOS only accepts one version (either 1 or 2) on a given interface.

## Usage Examples

The following example configures the virtual demand interface to accept only RIP version 2 packets:

(config)**#interface demand 1**
(config-demand 1)**#ip rip receive version 2**

# ip rip send version

Use the **ip rip send version** command to configure the RIP version the unit sends in all RIP packets transmitted on the interface. Use the **no** form of this command to restore the default value. Variations of this command include:

**ip rip send version 1**
**ip rip send version 2**

## Syntax Description

| | |
|---|---|
| **1** | Transmits only RIP version 1 packets on the interface. |
| **2** | Transmits only RIP version 2 packets on the interface. |

## Default Values

By default, all interfaces transmit RIP version 1 (the default value for the **version** command).

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 11.1 | Command expanded to include the demand interface. |

## Functional Notes

Use the **ip rip send version** to specify a RIP version that overrides the **version** (in the Router RIP) configuration.

The AOS only transmits one version (either 1 or 2) on a given interface.

## Usage Examples

The following example configures the virtual demand interface to transmit only RIP version 2 packets:

(config)**#interface demand 1**
(config-demand 1)**#ip rip send version 2**

# ip rip summary-address *<ip address> <subnet mask>*

Use the **ip rip summary-address** command to manually summarize the routes Routing Information Protocol (RIP) will advertise and send out a specified interface. Use the **no** form of this command to disable this mode.

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies the summarized network IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.0**). |
| *<subnet mask>* | Specifies the subnet mask that corresponds to the range of IP addresses (network). Subnet masks can be expressed in dotted decimal notation (for example, **255.255.255.0**) or as a prefix length (for example, **/24**). |

## Default Values

By default, no manual summarization is applied by RIP.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Functional Notes

Unlike the automatic summarization on classful network boundaries, only specific network advertisements are made by RIP using the **ip rip summary-address** command. This command is only effective if RIP version 2 is configured.

## Usage Examples

The following example enables manual summarization on the specified IP address:

(config)**#interface demand 1**
(config-demand 1)**#ip rip summary-address 10.10.123.0 255.255.255.0**

# ip route-cache

Use the **ip route-cache** command to enable fast-cache switching on the interface. Use the **no** form of this command to disable fast-cache switching and return to process switching mode.


*Using Network Address Translation (NAT) or the AOS firewall capabilities on an interface requires process switching mode (using the **no ip route-cache** command).*

## Syntax Description

No subcommands.

## Default Values

By default, fast-cache switching is enabled on all Ethernet and virtual Frame Relay sub-interfaces. IP route cache is enabled for all virtual demand interfaces.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |
| Release 11.1 | Command expanded to include the demand interface. |

## Functional Notes

Fast-cache switching allows an IP interface to provide optimum performance when processing IP traffic.

## Usage Examples

The following example enables fast-cache switching on the virtual demand interface:

(config)**#interface demand 1**
(config-demand 1)**#ip route-cache**

# ip unnumbered *<interface>*

Use the **ip unnumbered** command to use the IP address assigned to the specified interface for all IP processing on the active interface. Use the **no** form of this command to remove the unnumbered configuration.

## Syntax Description

| | |
|---|---|
| *<interface>* | Specifies the interface that contains the IP address to use as the source address for all packets transmitted on this interface. Specify an interface in the format *<interface type [slot/port \| slot/port.sub-interface id \| interface id \| interface id.sub-interface id]>*. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **show ip unnumbered ?** for a list of valid interfaces. |

## Default Values

By default, all interfaces are configured to use a specified IP address (using the **ip address** command).

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 11.1 | Command expanded to include the demand interface. |

## Functional Notes

If **ip unnumbered** is enabled on an interface, all IP traffic from the interface will use a source IP address taken from the specified interface. For example, specifying **ip unnumbered eth 0/1** while in the Demand Interface Configuration mode configures the demand interface to use the IP address assigned to the Ethernet interface for all IP processing. In addition, the AOS uses the specified interface information when sending route updates over the unnumbered interface. Static routes may either use the interface name (ppp 1) or the far-end address (if it will be discovered).

## Usage Examples

The following example configures the demand interface (labeled **demand 1**) to use the IP address assigned to the Ethernet interface (**eth 0/1**):

(config)**#interface demand 1**
(config-demand 1)**#ip unnumbered eth 0/1**

# ip urlfilter *<name>*

Use the **ip urlfilter** command to apply a URL filter to the interface for all inbound or outbound traffic. Use the **no** form of this command to remove the URL filter from an interface. Variations of this command include:

**ip urlfilter** <*name*> **in**
**ip urlfilter** <*name*> **out**

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies the URL filter name to use on the interface. |
| **in** | Applies the filter to the inbound traffic. |
| **out** | Applies the filter to the outbound traffic. |

## Default Values

By default, there are no URL filters applied to any interfaces.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Functional Notes

The firewall must be enabled using the **ip firewall** command in order to use URL filters. The URL filter must be created by using the **ip urlfilter *<filtername>* http** command before applying it to the interface. See *ip urlfilter <name> http* on page 598 for more information on using this command.

## Usage Example

The following example performs URL filtering on all traffic entering through the demand interface (labeled **demand 1**) and matches the URL filter named **MyFilter**:

(config)**#interface demand 1**
(config-demand 1)**#ip urlfilter MyFilter in**

# keepalive <*value*>

Use the **keepalive** command to enable the transmission of keepalive packets on the interface and specify the time interval in seconds between transmitted packets. Use the **no** form of this command to return to the default setting.

## Syntax Description

| | |
|---|---|
| <*value*> | Defines the time interval (in seconds) between transmitted keepalive packets. Range is 0 to 32,767 seconds. |

## Default Values

By default, the time interval between transmitted keepalive packets is 10 seconds.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 11.1 | Command expanded to include the demand interface. |

## Functional Notes

If three keepalive packets are sent to an interface with no response, the interface is considered down. To detect interface failures quickly, specify a smaller keepalive time.

## Usage Examples

The following example specifies a keepalive time of 5 seconds on the virtual demand interface:

(config)**#interface demand 1**
(config-demand 1)**#keepalive 5**

# lldp receive

Use the **lldp receive** command to allow LLDP packets to be received on this interface. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, all interfaces are configured to send and receive LLDP packets.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |
| Release 11.1 | Command expanded to include the demand interface. |

## Usage Examples

The following example configures the demand interface to receive LLDP packets:

(config)**#interface demand 1**
(config-demand 1)**#lldp receive**

# lldp send

Use the **lldp send** command to configure this interface to transmit LLDP packets or to control the types of information contained in the LLDP packets transmitted by this interface. Use the **no** form of these commands to disable these features. Variations of this command include:

**lldp send**
**lldp send management-address**
**lldp send port-description**
**lldp send system-capabilities**
**lldp send system-description**
**lldp send system-name**
**lldp send-and-receive**

## Syntax Description

| | |
|---|---|
| **management-address** | Enables transmission of management address information on this interface. |
| **port-description** | Enables transmission of port description information on this interface. |
| **system-capabilities** | Enables transmission of this device's system capabilities on this interface. |
| **system-description** | Enables transmission of this device's system description on this interface. |
| **system-name** | Enables transmission of this device's system name on this interface. |
| **and-receive** | Configures this interface to both transmit and receive LLDP packets. |

## Default Values

Be default, all interfaces are configured to transmit and receive LLDP packets of all types.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |
| Release 11.1 | Command expanded to include the demand interface. |

## Functional Notes

Individual LLDP information can be enabled or disabled using the various forms of the **lldp send** command. For example, use the **lldp send and-receive** command to enable transmit and receive of all LLDP information. Then use the **no lldp send port-description** command to prevent LLDP from transmitting port description information.

## Usage Examples

The following example configures the demand interface to transmit LLDP packets containing all enabled information types:

(config)**#interface demand 1**
(config-demand 1)**#lldp send**

The following example configures the demand interface to transmit and receive LLDP packets containing all information types:

(config)**#interface demand 1**
(config-demand 1)**#lldp send-and-receive**

# match-interesting

Use the **match-interesting** command to allow an access control list (ACL) to specify which traffic attempting to cross this interface will be considered interesting. Use the **no** form of this command to restore the default values. Variations of this command include:

**match-interesting list** <*name*>
**match-interesting list** <*name*> **in**
**match-interesting list** <*name*> **out**
**match-interesting reverse list** <*name*>
**match-interesting reverse list** <*name*> **in**
**match-interesting reverse list** <*name*> **out**

## Syntax Description

| | |
|---|---|
| **list** <*name*> | Specifies using an ACL with normal (source, destination) ACL matching logic. |
| **reverse list** <*name*> | Specifies using an ACL with reverse (destination, source) ACL matching logic. |
| **in** | Optional. Specifies that only incoming traffic is interesting. |
| **out** | Optional. Specifies that only outgoing traffic is interesting. |

## Default Values

By default, no interesting traffic is defined.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example instructs demand interface **1** to use the access control list **MyACL** when checking for interesting traffic:

(config)**#interface demand 1**
(config-demand 1)**#match-interesting list MyACL in**

# max-reserved-bandwidth <*value*>

Use the **max-reserved-bandwidth** command to specify the percentage of interface bandwidth reserved for use in user-defined (priority or class-based) queues. The remainder of the interface bandwidth is reserved for system critical traffic and is not available to user-defined queues. Use the **no** form of this command to restore the default values.


*Reserving a portion of the interface bandwidth for system critical traffic is necessary for proper operation. Specifying the entire interface bandwidth for use in user-defined queues can cause undesirable operation.*

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the maximum percentage of bandwidth to reserve for QoS. This setting is configured as a percentage of the total interface speed. Range is 1 to 100 percent. |

## Default Values

By default, **max-reserved-bandwidth** is set to 75 percent, which reserves 25 percent of the interface bandwidth for system critical traffic.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example specifies **85** percent of the bandwidth on the demand interface 1 be available for use in user-defined queues:

(config)**#interface demand 1**
(config-demand 1)**#max-reserved-bandwidth 85**

# mtu <*size*>

Use the **mtu** command to configure the maximum transmit unit (MTU) size for the active interface. Use the **no** form of this command to return to the default value.

## Syntax Description

| | | |
|---|---|---|
| <*size*> | Configures the window size for transmitted packets. The valid ranges for the various interfaces are listed below: | |
| | ATM interfaces | 64 to 1520 |
| | Demand interfaces | 64 to 1520 |
| | Ethernet interfaces | 64 to 1500 |
| | FDL interfaces | 64 to 256 |
| | HDLC interfaces | 64 to 1520 |
| | Loopback interfaces | 64 to 1500 |
| | Tunnel interfaces | 64 to 18,190 |
| | Virtual Frame Relay sub-interfaces | 64 to 1520 |
| | Virtual PPP interfaces | 64 to 1500 |

## Default Values

| | | |
|---|---|---|
| <*size*> | The default values for the various interfaces are listed below: | |
| | ATM interfaces | 1500 |
| | Demand interfaces | 1500 |
| | Ethernet interfaces | 1500 |
| | FDL interfaces | 256 |
| | HDLC interfaces | 1500 |
| | Loopback interfaces | 1500 |
| | Tunnel interfaces | 1500 |
| | Virtual Frame Relay sub-interfaces | 1500 |
| | Virtual PPP interfaces | 1500 |

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 11.1 | Command expanded to include the demand interface. |

## Functional Notes

OSPF will not become adjacent on links where the MTU sizes do not match. If router A and router B are exchanging hello packets but their MTU sizes do not match, they will never reach adjacency. This is by design and required by the RFC.

## Usage Examples

The following example specifies an MTU of 1200 on the virtual demand interface:

(config)**#interface demand 1**
(config-demand 1)**#mtu 1200**

# peer default ip address *<ip address>*

Use the **peer default ip address** command to specify the default IP address of the remote end of this interface. Use the **no** form of this command to remove a configured default IP address.

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies the default IP address for the remote end. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

## Default Values

By default, there is no assigned peer default IP address.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |
| Release 11.1 | Command expanded to include the demand interface. |

## Functional Notes

This command is useful if the peer does not send the IP address option during PPP negotiations.

## Usage Examples

The following example sets the default peer IP address to **192.22.71.50**:

(config)**#interface demand 1**
(config-demand 1)**#peer default ip address 192.22.71.50**

# ppp authentication

Use the **ppp authentication** command to specify the authentication protocol on the PPP virtual interface that the peer should use to authenticate itself. Use the **no** form of this command to remove configured PPP authentication. Variations of this command include:

**ppp authentication chap**
**ppp authentication pap**

## Syntax Description

| | |
|---|---|
| **chap** | Configures CHAP authentication on the interface. |
| **pap** | Configures PAP authentication on the interface. |

## Default Values

By default, PPP endpoints have no authentication configured.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 11.1 | Command expanded to include the demand interface. |

## Technology Review

CHAP and PAP are two authentication methods that enjoy widespread support. Both methods are included in the AOS and are easily configured.


*The authentication method set up on the local router can be different from that on the peer. Also, just because one router requires authentication from its peer does not mean it also has to authenticate itself to the peer.*

### Defining PAP

The Password Authentication Protocol (PAP) is used to verify that the PPP peer is a permitted device by checking a user name and password configured on the peer. The user name and password are both sent unencrypted across the connecting private circuit.

PAP requires two-way message passing. First, the router that is required to be authenticated (say the peer) sends an authentication request with its user name and password to the router requiring authentication (say the local router). The local router then looks up the user name and password in the user name database within the PPP interface, and if they match sends an authentication acknowledge back to the peer.


*The PPP user name and password database is separate and distinct from the global user name password database. For PAP and CHAP, use the database under the PPP interface configuration.*

Several example scenarios are given below for clarity.

**Configuring PAP Example 1: Only the local router requires the peer to authenticate itself.**

On the local router (host name Local):

Local(config-demand 1)**#ppp authentication pap**
Local(config-demand 1)**#username farend password same**

On the peer (host name Peer):

Peer(config-demand 1)**#ppp pap sent-username farend password same**

The first line of the configuration sets the authentication mode as PAP. This means the peer is required to authenticate itself to the local router via PAP. The second line is the user name and password expected to be sent from the peer. On the peer, the **ppp pap sent-username** command is used to specify the appropriate matching user name and password.

**Configuring PAP Example 2: Both routers require the peer to authenticate itself.**

On the local router (host name Local):

Local(config-demand 1)**#ppp authentication pap**
Local(config-demand 1)**#username farend password far**
Local(config-demand 1)**#ppp pap sent-username nearend password near**

On the peer (host name Peer):

Peer(config-demand 1)**#ppp authentication pap**
Peer(config-demand 1)**#username nearend password near**
Peer(config-demand 1)**#ppp pap sent-username farend password far**

Now both routers send the authentication request, verify that the user name and password sent match what is expected in the database, and send an authentication acknowledge.

**Defining CHAP**

The Challenge-Handshake Authentication Protocol (CHAP) is a three-way authentication protocol composed of a challenge response and success or failure. The MD5 protocol is used to protect user names and passwords in the response.

First, the local router (requiring its peer to be authenticated) sends a "challenge" containing only its own unencrypted user name to the peer. The peer then looks up the user name in the user name database within the PPP interface, and if found takes the corresponding password and its own host name and sends a “response” back to the local router. This data is encrypted. The local router verifies that the user name and password are in its own user name database within the PPP interface, and if so sends a "success" back to the peer.


*The PPP user name and password database is separate and distinct from the global user name password database. For PAP and CHAP, use the database under the PPP interface configuration.*

Several example scenarios are given below for clarity.

**Configuring CHAP Example 1: Only the local router requires the peer to authenticate itself.**

On the local router (host name Local):

Local(config-demand 1)**#ppp authentication chap**
Local(config-demand 1)**#username Peer password same**

On the peer (host name Peer):

Peer(config-demand 1)**#username Local password same**

The first line of this configuration sets the authentication mode to CHAP. This means the peer is required to authenticate itself to the local router via CHAP. The second line is the user name and password expected to be sent from the peer. The peer must also have the **username** up both to verify the incoming user name from the local router and to use the password (along with its host name) in the response to the local router.


*Both ends must have identical passwords.*

**Configuring CHAP Example 2: Both routers require the peer to authenticate itself.**

On the local router (host name Local):

Local(config-demand 1)**#ppp authentication chap**
Local(config-demand 1)**#username Peer password same**

On the peer (host name Peer):

Peer(config-demand 1)**#ppp authentication chap**
Peer(config-demand 1)**#username Local password same**

This is basically identical to Example 1 except that both routers will now challenge each other and respond.

**Configuring CHAP Example 3: Using the ppp chap hostname command as an alternate solution.**

On the local router (host name Local):

Local(config-demand 1)**#ppp authentication chap**
Local(config-demand 1)**#username Peer password same**
Local(config-demand 1)**#ppp chap hostname nearend**

On the peer (host name Peer):

Peer(config-demand 1)**#username nearend password same**

Notice the peer is expecting user name **nearend** even though the local router's host name is **Local**. Therefore the local router can use the **ppp chap hostname** command to send the correct name on the challenge.

**Configuring CHAP Example 4: Using the ppp chap password command as an alternate solution.**

On the local router (host name Local):

Local(config-demand 1)**#ppp authentication chap**
Local(config-demand 1)**#username Peer password different**

On the peer (host name Peer):

Peer(config-demand 1)**#username Local password same**
Peer(config-demand 1)**#ppp chap password different**

Here the local router challenges with host name **Local**. The peer verifies the name in the user name database, but instead of sending the password **same** in the response, it uses the one in the **ppp chap password** command. The local router then verifies that user **Peer** with password **different** is valid and sends a success message.

# ppp chap hostname *<name>*

Use the **ppp chap hostname** command to configure an alternate host name for CHAP PPP authentication. Use the **no** form of this command to remove a configured host name. For more information on PAP and CHAP functionality, refer to the *Technology Review* section for the command *ppp authentication* on page 1142.

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies a host name using an alphanumeric string up to 80 characters in length. |

## Default Values

By default, there are no configured PPP CHAP host names.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 11.1 | Command expanded to include the demand interface. |

## Usage Examples

The following example specifies a PPP CHAP host name of **my_host**:

(config)**#interface demand 1**
(config-demand 1)**#ppp chap hostname my_host**

# ppp chap password *<password>*

Use the **ppp chap password** command to configure an alternate password when the peer requires CHAP PPP authentication. Use the **no** form of this command to remove a configured password. For more information on PAP and CHAP functionality, refer to the *Technology Review* section for the command *ppp authentication* on page 1142.

## Syntax Description

| | |
|---|---|
| *<password>* | Specifies a password using an alphanumeric string up to 80 characters in length. |

## Default Values

By default, there is no defined PPP CHAP password.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 11.1 | Command expanded to include the demand interface. |

## Usage Examples

The following example specifies a PPP CHAP password of **my_password:**

(config)**#interface demand 1**
(config-demand 1)**#ppp chap password my_password**

# ppp multilink

Use the **ppp multilink** command to enable multilink PPP (MPPP) operation on an existing PPP interface. Use the **no** form of this command to disable. Variations of this command include:

**ppp multilink**
**ppp multilink fragmentation**
**ppp multilink interleave**
**ppp multilink maximum** <*number*>

## Syntax Description

| | |
|---|---|
| **fragmentation** | Enables multilink fragmentation operation. |
| **interleave** | Enables multilink interleave operation. |
| **maximum** <*number*> | Specifies the maximum number of links allowed in a PPP multilink bundle. |

## Default Values

By default, MPPP is disabled.

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |
| Release 7.2 | Fragmentation and interleave operation were added. |
| Release 11.1 | Command expanded to include the demand interface. |

## Functional Notes

When enabled, this interface is capable of the following:

- Combining multiple physical links into one logical link.
- Receiving upper layer protocol data units (PDU), fragmenting and transmitting over the physical links.
- Receiving fragments over the physical links and reassembling them into PDUs.

The fragmentation and interleave options can be used to enhance the multilink operation. Fragmentation is used to reduce serialization delays of large packets. The fragmentation process evenly divides the data among all links in the bundle with a minimum packet size of 96 bytes. The interleave operation is used with streaming protocols to reduce delay by giving priority to packets identified as high priority. In order delivery is guaranteed with multilink fragmentation, but is not guaranteed with multilink interleave operation.

The multilink bundle will remain active with a minimum of one physical link. Physical links may be dynamically added or removed from the multilink bundle with minor interruption to traffic flow.

## Usage Examples

The following example enables MPPP:

(config)**#interface demand 1**
(config-demand 1)**#ppp multilink**

# ppp pap sent-username <*username*> password <*password*>

Use the **ppp pap sent-username/password** command to configure a user name and password when the peer requires PAP PPP authentication. Use the **no** form of this command to remove a configured password. For more information on PAP and CHAP functionality, refer to the *Technology Review* section for the command *ppp authentication* on page 1142.

## Syntax Description

| | |
|---|---|
| <*username*> | Specifies a user name by alphanumeric string up to 80 characters in length (the user name is case-sensitive). |
| <*password*> | Specifies a password by alphanumeric string up to 80 characters in length (the password is case-sensitive). |

## Default Values

By default, there is no defined **ppp pap sent-username** and **password**.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 11.1 | Command expanded to include the demand interface. |

## Usage Examples

The following example specifies a PPP PAP sent-user name of **local** and a password of **my_password**:

(config)#**interface demand 1**
(config-demand 1)#**ppp pap sent-username local password my_password**

# qos-policy out *<name>*

Use the **qos-policy out** command to apply a previously-configured QoS map to outgoing packets on an interface. Use the **no** form of this command to remove the map from the interface.

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies the name of a previously-created QoS map (refer to *qos map <name> <number>* on page 643 for more information). |

## Default Values

No default value is necessary for this command.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |
| Release 11.1 | Command expanded to include the demand interface. |

## Usage Examples

The following example applies the QoS map **VOICEMAP** to the demand 1 interface:

(config)**#interface demand 1**
(config-demand 1)**#qos-policy out VOICEMAP**

# resource pool *<name>*

Use the **resource pool** command to associate a resource pool with the demand interface. No more than one resource pool may be associated with an interface. Refer to *resource pool-member <name>* on page 784 for more information. Use the **no** form of this command to restore the default values.

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies the resource pool that this interface will use to originate/answer demand connections. |

## Default Values

By default, no resource pool is associated with this interface.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example associates the resource pool named **Pool1** with demand interface **1**:

(config)**#interface demand 1**
(config-demand 1)**#resource pool Pool1**

# snmp trap link-status

Use the **snmp trap link-status** command to control the Simple Network Management Protocol (SNMP) variable ifLinkUpDownTrapEnable (RFC2863) to enable (or disable) the interface to send SNMP traps when there is an interface status change. Use the **no** form of this command to disable this trap.

## Syntax Description

No subcommands.

## Default Values

By default, the ifLinkUpDownTrapEnable OID is enabled for all interfaces except virtual Frame Relay interfaces.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 11.1 | Command expanded to include the demand interface. |

## Functional Notes

The **snmp trap link-status** command is used to control the RFC2863 ifLinkUpDownTrapEnable OID (OID number 1.3.6.1.2.1.31.1.1.1.14.0).

## Usage Examples

The following example disables the link-status trap on the virtual demand interface:

(config)**#interface demand 1**
(config-demand 1)**#no snmp trap link-status**

# **username** *<username>* **password** *<password>*

Use the **username password** command to configure the user name and password of the peer to use for demand authentication. Use the **no** form of this command to remove a configured user name and password.

## Syntax Description

| | |
|---|---|
| *<username>* | Specifies a user name by alphanumerical string up to 30 characters in length (the user name is case-sensitive). |
| *<password>* | Specifies a password by alphanumerical string up to 30 characters in length (the password is case-sensitive). |

## Default Values

By default, there is no established user name and password.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 11.1 | Command expanded to include the demand interface. |

## Functional Notes

PAP uses this entry to check received information from the peer. CHAP uses this entry to check the received peer host name and a common password.

## Usage Examples

The following example creates a user name of **ADTRAN** with password **ADTRAN** for the demand link labeled **5**:

(config)**#interface demand 5**
(config-demand 5)**#username ADTRAN password ADTRAN**

# FRAME RELAY INTERFACE CONFIG COMMAND SET

To activate the Frame Relay Interface Configuration mode, enter the **interface frame-relay** command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#interface frame-relay 1
(config-fr 1)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:



All other commands for this command set are described in this section in alphabetical order.



# bandwidth <*value*>

Use the **bandwidth** command to provide the bandwidth value of an interface to the higher-level protocols. This value is used in cost calculations. Use the **no** form of this command to restore the default values.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies bandwidth in kbps. |

## Default Values

No default value is necessary for this command.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Functional Notes

The **bandwidth** command is an informational value that is communicated to the higher-level protocols to be used in cost calculations. This is a routing parameter only and does not affect the physical interface.

## Usage Examples

The following example sets bandwidth of the Frame Relay interface to 10 Mbps:

(config)**#interface frame-relay 1**
(config-fr 1)**#bandwidth 10000**

# encapsulation frame-relay ietf

Use the **encapsulation frame-relay ietf** command to configure the encapsulation on a virtual Frame Relay interface as IETF (RFC1490). Currently, this is the only encapsulation setting. Settings for this option must match the far-end router's settings in order for the Frame Relay interface to become active.

## Syntax Description

No subcommands.

## Default Values

By default, all Frame Relay interfaces use IETF encapsulation.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example configures the endpoint for IETF encapsulation:

(config)**#interface frame-relay 1**
(config-fr 1)**#encapsulation frame-relay ietf**

# fair-queue

Use the **fair-queue** command to enable weighted fair queuing (WFQ) on an interface. Use the **no** form of this command to disable WFQ and enable first-in-first-out (FIFO) queueing for an interface. WFQ is enabled by default for WAN interfaces. Variations of this command include:

**fair-queue**
**fair-queue** <*value*>

## Syntax Description

| | |
|---|---|
| <*value*> | Optional. Specifies the maximum number of packets that can be present in each conversation sub-queue. Packets received for a conversation after this limit is reached are discarded. Range is 16 to 512 packets. |

## Default Values

By default, **fair-queue** is enabled with a threshold of 64 packets.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following example enables WFQ on the interface with a threshold set at 100 packets:

(config)**#interface frame-relay 1**
(config-fr 1)**#fair-queue 100**

# frame-relay intf-type

Use the **frame-relay intf-type** command to define the Frame Relay signaling role needed for the endpoint. Use the **no** form of this command to return to the default value. Variations of this command include:

**frame-relay intf-type dce**
**frame-relay intf-type dte**
**frame-relay intf-type nni**

## Syntax Description

| | |
|---|---|
| **dce** | Specifies DCE or network-signaling role. Use this interface type when you need the unit to emulate the frame switch. |
| **dte** | Specifies DTE or user-signaling role. Use this interface type when connecting to a Frame Relay switch (or piece of equipment emulating a frame switch). |
| **nni** | Configures the interface to support both network and user signaling (DTE or DCE) when necessary. |

## Default Values

By default, **frame-relay intf-type** is set to **dte**.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example configures the Frame Relay endpoint for DCE signaling:

(config)**#interface frame-relay 1**
(config-fr 1)**#frame-relay intf-type dce**

# frame-relay lmi-n391dce <*value*>

Use the **frame-relay lmi-n391dce** command to set the N391 full status polling counter for the DCE endpoint. Typical applications should leave the default value for this timer. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Sets the poll counter value. Valid range is 1 to 255. |

## Default Values

By default, the polling counter for the DCE endpoint is set to six polls.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

The N391 counter determines how many link integrity polls occur in between full status polls. The number of link integrity polls between full status polls is n - 1, where n represents the full status poll. n can be set to any number between 1 and 255, but the default is used for most applications.

## Usage Examples

The following example sets the N391 counter for three polls:

(config)**#interface frame-relay 1**
(config-fr 1)**#frame-relay lmi-n391dce 3**

# frame-relay lmi-n391dte <*value*>

Use the **frame-relay lmi-n391dte** command to set the N391 full status polling counter for the DTE endpoint. Typical applications should leave the default value for this timer. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Sets the poll counter value. Valid range is 1 to 255. |

## Default Values

By default, the polling counter for the DTE endpoint is set to six polls.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

The N391 counter determines how many link integrity polls occur in between full status polls. The number of link integrity polls between full status polls is n - 1, where n represents the full status poll. n can be set to any number between 1 and 255, but the default is used for most applications.

## Usage Examples

The following example sets the N391 counter for three polls:

(config)**#interface frame-relay 1**
(config-fr 1)**#frame-relay lmi-n391dte 3**

# frame-relay lmi-n392dce <*value*>

Use the **frame-relay lmi-n392dce** command to set the N392 error threshold for the DCE endpoint. Typical applications should leave the default value for this setting. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Sets the error threshold value. Valid range is 1 to 10 errors. |

## Default Values

By default, the error threshold for the DCE endpoint is set to three errors.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

If the error threshold is met, the signaling state status is changed to down, indicating a service-affecting condition. This condition is cleared once N393 consecutive error-free events are received. N392 defines the number of errors required in a given event window, while N393 defines the number of polling events in each window.

For example:

If N392 = and N393 = 4, then if three errors occur within any four events, the interface is determined inactive.

## Usage Examples

The following example sets the N392 threshold for 5 seconds:

(config)**#interface frame-relay 1**
(config-fr 1)**#frame-relay lmi-n392dce 5**

# frame-relay lmi-n392dte <*value*>

Use the **frame-relay lmi-n392dte** command to set the N392 error threshold for the DTE endpoint. Typical applications should leave the default value for this setting. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Sets the error threshold value. Valid range is 1 to 10 errors. |

## Default Values

By default, the error threshold for the DTE endpoint is set to three errors.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

If the error threshold is met, the signaling state status is changed to down, indicating a service-affecting condition. This condition is cleared once N393 consecutive error-free events are received. N392 defines the number of errors required in a given event window, while N393 defines the number of polling events in each window.

For example:

If N392 = 3 and N393 = 4, then if three errors occur within any four events, the interface is determined inactive.

## Usage Examples

The following example sets the N392 threshold for five errors:

(config)**#interface frame-relay 1**
(config-fr 1)**#frame-relay lmi-n392dte 5**

# frame-relay lmi-n393dce <*value*>

Use the **frame-relay lmi-n393dce** to set the N393 LMI monitored event counter for the DCE endpoint. Typical applications should leave the default value for this counter. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Sets the event counter value. Valid range is 1 to 10 events. |

## Default Values

By default, the LMI monitored event counter for the DCE endpoint is set to four events.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example sets the N393 threshold for five events:

(config)**#interface frame-relay 1**
(config-fr 1)**#frame-relay lmi-n393dce 5**

# frame-relay lmi-n393dte <*value*>

Use the **frame-relay lmi-n393dte** command to set the N393 LMI monitored event counter for the DTE endpoint. Typical applications should leave the default value for this counter. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Sets the event counter value. Valid range is 1 to 10 events. |

## Default Values

By default, the LMI monitored event counter for the DTE endpoint is set to four events.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example sets the N393 threshold for five events:

(config)**#interface frame-relay 1**
(config-fr 1)**#frame-relay lmi-n393dte 5**

# frame-relay lmi-t391dte <*value*>

Use the **frame-relay lmi-t391dte** command to set the T391 signal polling timer for the DTE endpoint. Typical applications should leave the default value for this timer. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Sets the signal polling timer value in seconds. Valid range is 5 to 30 seconds. |

## Default Values

By default, the signal polling timer for the DTE endpoint is set to 10 seconds.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

The T391 timer sets the time (in seconds) between polls to the Frame Relay network.

## Usage Examples

The following example sets the T391 timer for 15 seconds:

(config)**#interface frame-relay 1**
(config-fr 1)**#frame-relay lmi-t391dte 15**

# frame-relay lmi-t392dce <*value*>

Use the **frame-relay lmi-t392dce** command to set the T392 polling verification timer for the DCE endpoint. Typical applications should leave the default value for this timer. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Sets the polling verification timer value in seconds. Valid range is 5 to 30 seconds. |

## Default Values

By default, the polling verification timer for the DCE endpoint is set to 10 seconds.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

The T392 sets the timeout (in seconds) between polling intervals. This parameter needs to be a few seconds longer than the T391 setting of the attached Frame Relay device.

## Usage Examples

The following example sets the T392 timer for 15 seconds:

(config)**#interface frame-relay 1**
(config-fr 1)**#frame-relay lmi-t392dce 15**

# frame-relay lmi-type

Use the **frame-relay lmi-type** command to define the Frame Relay signaling (LMI) type. Use the **no** form of the command to return to the default value. Variations of this command include:

**frame-relay lmi-type ansi**
**frame-relay lmi-type auto**
**frame-relay lmi-type cisco**
**frame-relay lmi-type none**
**frame-relay lmi-type q933a**

## Syntax Description

| | |
|---|---|
| **ansi** | Specifies Annex D signaling method. |
| **auto** | Automatically determines signaling type by messages received on the frame circuit. |
| **cisco** | Specifies Group of 4 signaling method. |
| **none** | Turns off signaling on the endpoint. This is used for dial-backup connections to ADTRAN IQ and Express series products. |
| **q933a** | Specifies Annex A signaling method. |

## Default Values

By default, the Frame Relay signaling type is set to **ansi**.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 2.1 | Added signaling type **none** to provide support for dial-backup to ADTRAN IQ and Express series products. |

## Usage Examples

The following example sets the signaling method for the endpoint to **cisco**:

(config)**#interface frame-relay 1**
(config-fr 1)**#frame-relay lmi-type cisco**

# frame-relay multilink

Use the **frame-relay multilink** command to enable the Frame Relay multilink interface. When the **no** form of this command is issued, all configuration options associated with this command and cross-connects made to this interface are removed. Variations of this command include:

**frame-relay multilink**
**frame-relay multilink ack** <*value*>
**frame-relay multilink bandwidth-class [A | B]**
**frame-relay multilink bandwidth-class C** <*threshold*>
**frame-relay multilink bid** <*string*>
**frame-relay multilink hello** <*value*>
**frame-relay multilink retry** <*number*>

## Syntax Description

| | |
|---|---|
| **ack** <*value*> | Optional. Specifies a wait for acknowledgement time (in seconds) for every bundle link in the bundle. Range is 1 to 180 seconds. |
| **bandwidth-class** | Optional. Specifies the class of operation, placing a minimum limit on the acceptable amount of bandwidth required for a bundle to up. |
| **[A \| B \| C]** | Optional. Specifies the class of operation.<br>Class A A single active link is sufficient for the bundle to be up.<br>Class B All defined bundle links must be active for the bundle to be up.<br>Class C A minimum threshold of links must be active for the bundle to be up. |
| <*threshold*> | Optional. Specifies the minimum number of active bundle links required for a class C bundle to be in the up state. This option will not be available unless Class C is specified. Range is 1 to 65,535 links. |
| **bid** <*string*> | Optional. Specifies a bundle ID (up to 48 characters) for the multilink bundle. All hello messages sent on links belonging to the multilink bundle contain the bundle ID. By default, the AOS creates a generic bundle ID for each configured multilink bundle using the following: **MFR** <*interface number*> where the interface number corresponds to the interface number of the Frame Relay interface. For example, if multilink is enabled on frame-relay interface 1, by default the bundle ID is **MFR1**. Changing the bundle ID causes the multilink connection to go down for renegotiation. |
| **hello** <*value*> | Optional. Specifies the time (in seconds) between hello messages for every bundle link in the bundle. Range is 1 to 180 seconds. |
| **retry** <*number*> | Optional. Specifies the number of times a bundle link will retransmit a message while waiting for acknowledgement. Range is 1 to 5 times. |

## Default Values

The default **ack** value is 4 seconds. The default **hello** value is 10 seconds. The default <*class*> value is A. The default **retry** value is 2.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Functional Note

This command is different from **ppp multilink**. In **ppp multilink**, if multiple cross-connects are configured for the PPP interface without multilink PPP being enabled, the first link to bring up LCP will be the only link actually cross-connected. In Frame Relay multilink, since there is no protocol corresponding to LCP, all cross-connects will be removed and the user will be free to re-issue any cross-connect.

## Usage Examples

The following example enables the Frame Relay multilink interface and sets the time between **hello** messages to **45** seconds:

(config)**#interface frame-relay 1**
(config-fr 1)**#frame-relay multilink hello 45**

The following example specifies Class B operation:

(config)**#interface frame-relay 1**
(config-fr 1)**#frame-relay multilink bandwidth-class b**

The following example specifies Class C operation with a threshold of **5**:

(config)**#interface frame-relay 1**
(config-fr 1)**#frame-relay multilink bandwidth-class c 5**

# hold-queue <*value*> out

Use the **hold-queue out** command to change the overall size of an interface's WAN output queue. Use the **no** form of this command to return to the default settings.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the total number of packets the output queue can contain before packets are dropped. Range is 16 to 1000 packets. |

## Default Values

The default queue size for WFQ is 400. The default queue size for PPP FIFO and Frame Relay round robin is 200.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following example sets the overall output queue size to 700:

(config)**#interface frame-relay 1**
(config-fr 1)**#hold-queue 700 out**

# max-reserved-bandwidth <*value*>

Use the **max-reserved-bandwidth** command to specify the percentage of interface bandwidth reserved for use in user-defined (priority or class-based) queues. The remainder of the interface bandwidth is reserved for system critical traffic and is not available to user-defined queues. Use the **no** form of this command to restore the default values.


*Reserving a portion of the interface bandwidth for system critical traffic is necessary for proper operation. Specifying the entire interface bandwidth for use in user-defined queues can cause undesirable operation.*

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the maximum percentage of bandwidth to reserve for QoS. This setting is configured as a percentage of the total interface speed. Range is 1 to 100 percent. |

## Default Values

By default, **max-reserved-bandwidth** is set to 75 percent, which reserves 25 percent of the interface bandwidth for system critical traffic.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example specifies **85** percent of the bandwidth on the Frame Relay 1 interface to be available for use in user-defined queues:

(config)**#interface frame-relay 1**
(config-fr 1)**#max-reserved-bandwidth 85**

# qos-policy out *<name>*

Use the **qos-policy out** command to apply a previously-configured QoS map to outgoing packets on an interface. Use the **no** form of this command to remove the map from the interface.

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies the name of a previously-created QoS map (refer to *qos map <name> <number>* on page 643 for more information). |

## Default Values

No default value is necessary for this command.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Functional Notes

When a QoS policy is applied to an interface, it may be disabled if the interface bandwidth is not adequate to support the requested bandwidth on the map set. Once the bandwidth problem is resolved, the map will work again. The bandwidth will be rechecked on any of the following changes:

1. A priority or class-based entry is added to, deleted from, or changed in a QoS map set.
2. The interface bandwidth is changed by the **bandwidth** command on the interface.
3. A QoS policy is applied to an interface.
4. A cross-connect is created that includes an interface with a QoS policy.
5. The interface queuing method is changed to fair-queue to use weighted fair queuing.
6. The interface operational status changes.
7. The interface bandwidth changes for other reasons (e.g., when ADSL finishes training).

In order to prevent the map from being disabled in cases of temporary inadequate bandwidth (e.g., a single link goes down in a dual T1 multilink configuration where the map requests more than one T1's worth of bandwidth), the QoS map uses the maximum theoretical bandwidth on an interface, not the actual bandwidth at that time. This actually helps QoS keep higher priority class-based traffic working better than best-effort traffic when the bandwidth drops.

## Usage Examples

The following example applies the QoS map **VOICEMAP** to the Frame Relay interface:

(config)**#interface frame-relay 1**
(config-fr 1)**#qos-policy out VOICEMAP**

# snmp trap

Use the **snmp trap** command to enable all supported Simple Network Management Protocol (SNMP) traps on the interface. Use the **no** form of this command to disable this trap.

## Syntax Description

No subcommands.

## Default Values

By default, all interfaces (except virtual Frame Relay interfaces and sub-interfaces) have SNMP traps enabled.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 3.1 | Command was extended to the SHDSL interface. |
| Release 5.1 | Command was expanded to include Ethernet sub-interfaces and Gigabit Ethernet interfaces. |

## Usage Examples

The following example enables SNMP on the virtual Frame Relay interface:

(config)**#interface frame-relay 1**
(config-fr 1)**#snmp trap**

# snmp trap link-status

Use the **snmp trap link-status** command to control the Simple Network Management Protocol (SNMP) variable ifLinkUpDownTrapEnable (RFC2863) to enable (or disable) the interface to send SNMP traps when there is an interface status change. Use the **no** form of this command to disable this trap.

## Syntax Description

No subcommands.

## Default Values

By default, the ifLinkUpDownTrapEnable OID is enabled for all interfaces except virtual Frame Relay interfaces.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 3.1 | Command was extended to the SHDSL interface. |
| Release 5.1 | Command was expanded to include Gigabit Ethernet, port channel, VLAN, E1, and G.703 interfaces. |

## Functional Notes

The **snmp trap link-status** command is used to control the RFC2863 ifLinkUpDownTrapEnable OID (OID number 1.3.6.1.2.1.31.1.1.1.14.0).

## Usage Examples

The following example disables the link-status trap on the Frame Relay interface:

(config)**#interface frame-relay 1**
(config-fr 1)**#no snmp trap link-status**

# FRAME RELAY SUB-INTERFACE CONFIG COMMAND SET

To activate the Frame Relay Sub-Interface Configuration mode, enter the **interface frame-relay** command (and specify a sub-interface) at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#interface frame-relay 1.16
(config-fr 1.16)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:

*cross-connect* *on page 29*
*description <text>* *on page 32*
*do* *on page 33*
*end* *on page 34*
*exit* *on page 35*
*shutdown* *on page 36*

All other commands for this command set are described in this section in alphabetical order.

*access-policy <name>* *on page 1176*
*bandwidth <value>* *on page 1177*
*bridge-group <number>* *on page 1178*
*crypto map <name>* *on page 1179*
*dial-backup commands* *begin on page 1181*
*dynamic-dns* *on page 1198*
*frame-relay commands* *begin on page 1200*
*ip commands* *begin on page 1204*
*lldp receive* *on page 1238*
*lldp send* *on page 1239*
*media-gateway ip* *on page 1241*
*mtu <size>* *on page 1242*
*spanning-tree commands* *begin on page 1243*

# access-policy *<name>*

Use the **access-policy** command to assign a specified access policy to an interface. Access policies are applied to traffic entering an interface. Use the **no** form of this command to remove an access policy association. For more information on using access policies, refer to *ip policy-class <name>* on page 549.

> NOTE *Configured access policies will only be active if the **ip firewall** command has been entered at the Global Configuration mode prompt to enable the AOS security features. All configuration parameters are valid, but no security data processing will be attempted unless the security features are enabled.*

## Syntax Description

| | |
|---|---|
| *<name>* | Identifies the configured access policy by alphanumeric descriptor (all access policy descriptors are case-sensitive). |

## Default Values

By default, there are no configured access policies associated with an interface.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include Ethernet sub-interfaces. |
| Release 6.1 | Command was expanded to include VLAN interfaces. |

## Functional Notes

To assign an access policy to an interface, enter the interface configuration mode for the desired interface and enter **access-policy** *<policy name>*.

## Usage Examples

The following example associates the access policy **Private** (to allow inbound traffic to the Web server) to the Frame-Relay sub-interface 1.16:

Enable the AOS security features:
(config)**#ip firewall**

Associate the access policy with the Frame Relay sub-interface 1.16:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#access-policy Private**

# bandwidth *<value>*

Use the **bandwidth** command to provide the bandwidth value of an interface to the higher-level protocols. This value is used in cost calculations. Use the **no** form of this command to restore the default values.

## Syntax Description

| | |
|---|---|
| *<value>* | Specifies bandwidth in kbps. Range is 1 to 4,294,967,295 kbps. |

## Default Values

To view default values use the **show interfaces** command.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Functional Notes

The **bandwidth** command is an informational value that is communicated to the higher-level protocols to be used in cost calculations. This is a routing parameter only and does not affect the physical interface.

## Usage Examples

The following example sets bandwidth of the Frame Relay interface to 10 Mbps:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#bandwidth 10000**

# bridge-group *<number>*

Use the **bridge-group** command to assign an interface to the specified bridge group. This command is supported on all Ethernet interfaces, PPP virtual interfaces, and Frame Relay virtual sub-interfaces. Use the **no** form of this command to remove the interface from the bridge group.

## Syntax Description

| | |
|---|---|
| *<number>* | Specifies the bridge group number. Range is 1 to 255. |

## Default Values

By default, there are no configured bridge groups.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

A bridged network can provide excellent traffic management to reduce collisions and limit the amount of bandwidth wasted with unnecessary transmissions when routing is not necessary. Any two interfaces can be bridged (Ethernet to T1 bridge, Ethernet to Frame Relay sub-interface).

## Usage Examples

The following example assigns the Frame Relay sub-interface labeled 1.16 to bridge group 1:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#bridge-group 1**

# crypto map *<name>*

Use the **crypto map** command to associate crypto maps with the interface. Use the **no** form of this command to remove a crypto map from an interface.

> NOTE *When you apply a map to an interface, you are applying all crypto maps with the given map name. This allows you to apply multiple crypto maps if you have created maps which share the same name but have different map index numbers.*

> NOTE *For VPN configuration example scripts, refer to the technical support note **Configuring VPN** located on the ADTRAN OS System Documentation CD provided with your unit.*

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies the crypto map name that you wish to assign to the interface. |

## Default Values

By default, no crypto maps are assigned to an interface.

## Command History

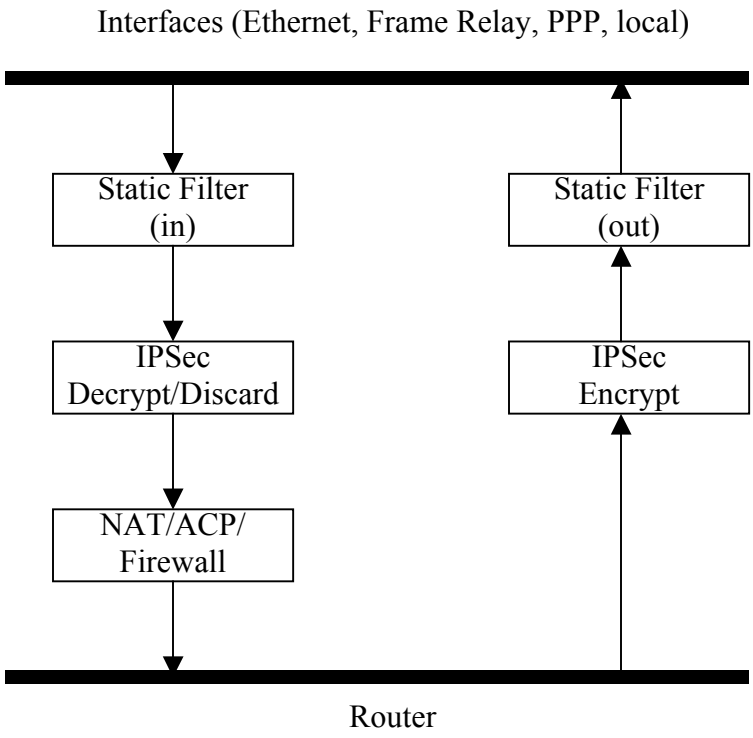
| | |
|---|---|
| Release 4.1 | Command was introduced. |

## Functional Notes

When configuring a system to use both the stateful inspection firewall and IKE negotiation for VPN, keep the following notes in mind.

When defining the policy-class and associated access-control lists (ACLs) that describe the behavior of the firewall, do not forget to include the traffic coming into the system over a VPN tunnel terminated by the system. The firewall should be set up with respect to the unencrypted traffic that is destined to be sent or received over the VPN tunnel. The following diagram represents typical AOS data-flow logic.




As shown in the diagram above, data coming into the product is first processed by the static filter associated with the interface on which the data is received. This access group is a true static filter and is available for use regardless of whether the firewall is enabled or disabled. Next (if the data is encrypted) it is sent to the IPSec engine for decryption. The decrypted data is then processed by the stateful inspection firewall. Therefore, given a terminating VPN tunnel, only unencrypted data is processed by the firewall.

The ACLs for a crypto map on an interface work in reverse logic to the ACLs for a policy class on an interface. When specifying the ACLs for a crypto map, the source information is the private local side, unencrypted source of the data. The destination information will be the far end, unencrypted destination of the data. However, ACLs for a policy class work in reverse. The source information for the ACL in a policy class is the far end. The destination information is the local side.

## Usage Examples

The following example applies all crypto maps with the name **MyMap** to the Frame Relay interface:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#crypto map MyMap**

# dial-backup auto-backup

Use the **dial-backup auto-backup** command to configure the interface to automatically attempt a dial-backup upon failure. Use the **no** form of this command to disable this feature. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of *dial-backup call-mode* on page 1184.

## Syntax Description

No subcommands.

## Default Values

By default, all backup endpoints will automatically attempt dial-backup upon a failure.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example enables automatic dial-backup on the endpoint:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#dial-backup auto-backup**

# dial-backup auto-restore

Use the **dial-backup auto-restore** command to configure the interface to automatically discontinue dial-backup when all network conditions are operational. Use the **no** form of this command to disable the auto-restore feature. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of *dial-backup call-mode* on page 1184.

## Syntax Description

No subcommands.

## Default Values

By default, all backup endpoints will automatically restore the primary connection when the failure condition clears.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example configures the AOS to automatically restore the primary connection when the failure condition clears:

```
(config)#interface frame-relay 1.16
(config-fr 1.16)#dial-backup auto-restore
```

# dial-backup backup-delay <*value*>

Use the **dial-backup backup-delay** command to configure the amount of time the router will wait after the failure condition is recognized before attempting to backup the link. Use the **no** form of this command to return to the default value. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of *dial-backup call-mode* on page 1184.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the delay period (in seconds) a failure must be active before the AOS will enter backup operation on the interface. Range is 10 to 86,400 seconds. |

## Default Values

By default, the **dial-backup backup-delay** period is set to 10 seconds.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example configures the AOS to wait **60** seconds (on an endpoint with an active alarm condition) before attempting dial-backup operation:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#dial-backup backup-delay 60**

# dial-backup call-mode

Use the **dial-backup call-mode** command to specify whether the configured backup interface answers or originates (or a combination of both ) backup calls. Use the **no** form of this command to return to the default value. Variations of this command include:

**dial-backup call-mode answer**
**dial-backup call-mode answer-always**
**dial-backup call-mode originate**
**dial-backup call-mode originate-answer**
**dial-backup call-mode originate-answer-always**

## Syntax Description

| | |
|---|---|
| **answer** | Answers and backs up primary link on failure. |
| **answer-always** | Answers and backs up regardless of primary link state. |
| **originate** | Originates backup call on primary link failure. |
| **originate-answer** | Originates or answers call on primary link failure. |
| **originate-answer-always** | Originates on failure; answers and backs up always. |

## Default Values

By default, the **dial-backup call-mode** is set to **originate-answer.**

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

The majority of the configuration for AOS dial-backup implementation is configured via the dial-backup PPP interface configuration commands. However, the numbers dialed are configured in the primary interface. Full sample configurations follow:

**Sample configuration for remote router (dialing out)**

```
hostname “Remote3200”
enable password adtran
!
interface eth 0/1
 ip address 192.168.1.254 255.255.255.0
 no shutdown
!
interface modem 1/3
 no shutdown
!
interface t1 1/1
 coding b8zs
```

```
 framing esf
 clock source line
 tdm-group 1 timeslots 1-24
 no shutdown
!
interface fr 1 point-to-point
 frame-relay lmi-type ansi
 no shutdown
 cross-connect 1 t1 1/1 1 fr 1
!
interface fr 1.16 point-to-point
 frame-relay interface-dlci 16
 ip address 10.1.1.2 255.255.255.252
 dial-backup call-mode originate
 dial-backup number 5551111 analog ppp1
 dial-backup number 5552222 analog ppp1
  no shutdown
!
interface ppp 1
 ip address 172.22.56.1 255.255.255.252
 ppp authentication chap
 username remoterouter password remotepass
 ppp chap hostname localrouter
 ppp chap password adtran
 no shutdown
!
ip route 192.168.2.0 255.255.255.0 172.22.56.2 255.255.255.252
!
line telnet 0 4
 password password
```

**Sample configuration for central router (dialing in)**

```
hostname “Central3200”
enable password adtran
!
interface eth 0/1
 ip address 192.168.100.254 255.255.255.0
 no shutdown
!
interface modem 1/3
 no shutdown
!
interface t1 1/1
 coding b8zs
```

```
 framing esf
 clock source line
 tdm-group 1 timeslots 1-24
 no shutdown
!
interface fr 1 point-to-point
 frame-relay lmi-type ansi
 no shutdown
 cross-connect 1 t1 1/1 1 fr 1
!
interface fr 1.100 point-to-point
 frame-relay interface-dlci 100
 ip address 10.1.1.1 255.255.255.252
 dial-backup call-mode answer
 dial-backup number 555-8888 analog ppp 1
!
interface ppp 1
 ip address 172.22.56.2 255.255.255.252
 ppp authentication chap
 username localrouter password adtran
 ppp chap hostname remoterouter
 ppp chap password remotepass
 no shutdown
!
ip route 192.168.1.0 255.255.255.0 172.22.56.1 255.255.255.252

line telnet 0 4
 password password
```

## Usage Examples

The following example configures AOS to generate backup calls for this endpoint using an analog modem interface (to phone number 555 1111) but never answer calls and specifies **ppp 1** as the backup interface:

```
(config)#interface frame-relay 1.16
(config-fr 1.16)#backup call-mode originate
(config-fr 1.16)#backup number 555 1111 analog ppp 1
```

## Technology Review

This technology review provides information regarding specific dial-backup router behavior (i.e., when the router will perform dial-backup, where in the configuration the AOS accesses specific routing information, etc.):

### Dialing Out

1. The AOS determines to place an outbound call when either the Layer 1 or Layer 2 has a failure.
2. When placing outbound calls, the AOS matches the number dialed to a PPP interface. This is accomplished with an addition to the **dial-backup number** command (refer to *dial-backup number <number>* on page 1191).
3. When placing the call, the AOS uses the configuration of the related PPP interface for authentication and IP negotiation.
4. If the call fails to connect on the first number dialed, the AOS places a call to the second number (if a second number is configured). The second number to be dialed references a separate PPP interface.

### Dialing In

1. The AOS receives an inbound call on a physical interface.
2. Caller ID is used to match the **dial-backup number** command to the configured PPP interface.
3. If a match is found, the call connects and the AOS pulls down the primary connection if it is not already in a down state.
4. If no match is found from caller ID, the call is terminated.

# dial-backup connect-timeout <*value*>

Use the **dial-backup connect-timeout** command to specify the number of seconds to wait for a connection after a call is attempted before trying to call again or dialing a different number. It is recommended this number be greater than 60. Use the **no** form of this command to return to the default setting. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of *dial-backup call-mode* on page 1184.

## Syntax Description

| | |
|---|---|
| <*value*> | Selects the amount of time (in seconds) that the router will wait for a connection before attempting another call. Valid range is 10 to 300 seconds. |

## Default Values

By default, the **dial-backup connect-timeout** period is set to 60 seconds.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example configures the AOS to wait **120** seconds before retrying a failed dial-backup call:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#dial-backup connect-timeout 120**

# dial-backup force

Use the **dial-backup force** command to manually override the automatic dial-backup feature. This can be used to force a link into backup to allow maintenance to be performed on the primary link without disrupting data. Use the **no** form of this command to return to the normal dial-backup operation state. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of *dial-backup call-mode* on page 1184. Variations of this command include:

**dial-backup force backup**
**dial-backup force primary**

## Syntax Description

| | |
|---|---|
| **backup** | Force backup regardless of primary link state. |
| **primary** | Force primary link regardless of its state. |

## Default Values

By default, this feature is disabled.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example configures the AOS to force this interface into dial-backup:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#dial-backup force backup**

# dial-backup maximum-retry <*value*>

Use the **dial-backup maximum-retry** command to select the number of calls the router will make when attempting to backup a link. Use the **no** form of this command to return to the default state. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of *dial-backup call-mode* on page 1184.

## Syntax Description

| | |
|---|---|
| <*value*> | Selects the number of call retries that will be made after a link failure. Valid range is 0 to 15 attempts.<br>Setting this value to 0 will allow unlimited retries during the time the network is failed. |

## Default Values

By default, **dial-backup maximum-retry** is set to 0 attempts.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example configures the AOS to retry a dial-backup call four times before considering backup operation not available:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#dial-backup maximum-retry 4**

# dial-backup number <*number*>

Use the **dial-backup number** command to configure the phone number and the call type the router will dial upon network failure. Multiple entries can be made for an interface to allow alternate sites to be dialed. Use the **no** form of this command to disable this feature. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of the command *dial-backup call-mode* on page 1184.Variations of this command include:

**dial-backup number** <*number*> **analog ppp** <*interface*>
**dial-backup number** <*number*> **digital-56k** <*isdn min chan*> <*isdn max chan*> **ppp** <*interface*>
**dial-backup number** <*number*> **digital 64k** <*isdn min chan*> <*isdn max chan*> **ppp** <*interface*>

## Syntax Description

| | |
|---|---|
| <*number*> | Specifies the phone numbers to call when the backup is initiated. |
| **analog ppp** | Indicates the number connects to an analog modem. |
| **digital-56k** | Indicates the number belongs to a digital 56 kbps per DS0 connection. |
| **digital-64k** | Indicates the number belongs to a digital 64 kbps per DS0 connection. |
| <*isdn min chan*> | Specifies the minimum number of DS0s required for a digital 56 or 64 kbps connection. Range is 1 to 24 DS0s. |
| <*isdn max chan*> | Specifies the maximum number of DS0s desired for a digital 56 or 64 kbps connection. Range is 1 to 24 DS0s. |
| **ppp** <*interface*> | Specifies the PPP interface to use as the backup for this interface. For example, **ppp 1**. |

## Default Values

By default, there are no configured dial-backup numbers.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example configures the AOS to dial 704-555-1212 (digital 64 kbps connection) to initiate dial-backup operation for this endpoint using the configured **ppp 1** backup interface:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#dial-backup number 7045551212 digital-64k 1 1 ppp 1**

# dial-backup priority <*value*>

Use the **dial-backup priority** command to select the backup priority for this interface. This command allows the user to establish the highest priority backup link and ensure that link will override backups attempted by lower priority links. Use the **no** form of this command to return to the default value. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of *dial-backup call-mode* on page 1184.

## Syntax Description

| | |
|---|---|
| <*value*> | Sets the relative priority of this link. Valid range is 0 to 100. A value of 100 designates the highest priority. |

## Default Values

By default, **dial-backup priority** is set to 50.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example assigns the highest priority to this endpoint:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#dial-backup priority 100**

# dial-backup randomize-timers

Use the **dial-backup randomize-timers** command to randomize the call timers to minimize potential contention for resources. Use the **no** form of this command to return to the default value. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of *dial-backup call-mode* on page 1184.

## Syntax Description

No subcommands.

## Default Values

By default, the AOS does not randomize the dial-backup call timers.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example configures the AOS to randomize the dial-backup timers associated with this endpoint:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#dial-backup randomize-timers**

# dial-backup redial-delay *<value>*

Use the **dial-backup redial-delay** command to configure the delay after an unsuccessful call until the call will be re-tried. Use the **no** form of this command to return to the default setting. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of *dial-backup call-mode* on page 1184.

## Syntax Description

| | |
|---|---|
| *<value>* | Specifies the delay in seconds between attempting to re-dial a failed backup attempt. Range is 10 to 3600 seconds. |

## Default Values

By default, **dial-backup redial-delay** is set to 10 seconds.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example configures a redial delay of **25** seconds on this endpoint:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#dial-backup redial-delay 25**

# dial-backup restore-delay *<value>*

Use the **dial-backup restore-delay** command to configure the amount of time the router will wait after the network is restored before disconnecting the backup link and reverting to the primary. This setting is used to prevent disconnecting the backup link if the primary link is "bouncing" in and out of alarm. Use the **no** form of this command to return to the default setting. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of *dial-backup call-mode* on page 1184.

## Syntax Description

| | |
|---|---|
| *<value>* | Specifies the number of seconds the AOS will wait (after a primary link is restored) before disconnecting dial-backup operation. Range is 10 to 86,400 seconds. |

## Default Values

By default, **dial-backup restore-delay** is set to 10 seconds.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example configures the AOS to wait **30** seconds before disconnecting dial-backup operation and restoring the primary connection for this endpoint:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#dial-backup restore-delay 30**

# dial-backup schedule

Use the **dial-backup schedule** command to set the time of day that backup will be enabled. Use this command if backup is desired only during normal business hours and on specific days of the week. Use the **no** form of this command to disable dial-backup (as specified). For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of *dial-backup call-mode* on page 1184. Variations of this command include:

**dial-backup schedule day** <*name*>
**dial-backup schedule enable-time** <*value*>
**dial-backup schedule disable-time** <*value*>

## Syntax Description

| | |
|---|---|
| **day** <*name*> | Sets the days to allow backup. Valid range is Monday through Sunday. |
| **enable-time** <*value*> | Sets the time of day to enable backup. Time is entered in 24-hour format (00:00). |
| **disable-time** <*value*> | Sets the time of day to disable backup. Time is entered in 24-hour format (00:00). |

## Default Values

By default, dial-backup is enabled for all days and times if the dial-backup auto-backup command has been issued and the dial-backup schedule has not been entered.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example enables dial-backup Monday through Friday 8:00 am to 7:00 pm:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#dial-backup schedule enable-time 08:00**
(config-fr 1.16)**#dial-backup schedule disable-time 19:00**
(config-fr 1.16)**#no dial-backup schedule day Saturday**
(config-fr 1.16)**#no dial-backup schedule day Sunday**

# dial-backup shutdown

Use the **dial-backup shutdown** command to deactivate all dial-backup functionality in the unit. Dial-backup configuration parameters are kept intact, but the unit will not initiate (or respond) to dial-backup sequences in the event of a network outage. Use the **no** form of this command to reactivate the dial-backup interface. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of *dial-backup call-mode* on page 1184.

## Syntax Description

No subcommands.

## Default Values

By default, all AOS interfaces are disabled.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example deactivates the configured dial-backup interface:

```
(config)#interface frame-relay 1.16
(config-fr 1.16)#dial-backup shutdown
```

# dynamic-dns

Use the **dynamic-dns** command to configure Dynamic DNS service provided by Dynamic Network Services, Inc. (www.dyndns.org). Use the **no** versions of these commands to disable these features. Variations of this command include:

**dynamic-dns custom** <*hostname*> <*minutes*>
**dynamic-dns dyndns** <*hostname*> <*username*> <*password*>
**dynamic-dns dyndns-custom** <*hostname*> <*username*> <*password*>
**dynamic-dns dyndns-static** <*hostname*> <*username*> <*password*>

## Syntax Description

| | |
|---|---|
| <*hostname*> | Specifies the host name for the server that updates the Dynamic Domain Name Server (DNS). |
| <*minutes*> | Specifies the intervals in minutes to update the server with information (updates also occur when the interface's IP address changes regardless of the update intervals).<br>Refer to *Functional Notes* below for additional argument descriptions. |

## Default Values

No default is necessary for this command.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |
| Release 12.1 | Command was expanded. |

## Functional Notes

**custom** - Constanttime.com's Custom Dynamic DNS$^{SM}$ service allows you complete access and management control over your domain name regardless of where you purchased/registered it. This allows to manage IP address mappings (A records), domain aliases (CNAME records) and mail servers (MX records).

**dyndns** - The Dynamic DNS$^{SM}$ offered by Dynamic Network Services Inc., (DynDNS.org) allows you to alias a dynamic IP address to a static host name in various domains. This allows your unit to be more easily accessed from various locations on the Internet. This service is provided for up to five host names.

**dyndns-custom -** DynDNS.org's Custom DNS$^{SM}$ service provides a full DNS solution, giving you complete control over an entire domain name. A web-based interface provides two levels of control over your domain, catering to average or advanced users. Five globally redundant DNS servers ensure that your domain will always resolve.

A choice of two interfaces is available. The basic interface is designed for most users. It comes preconfigured for the most common configuration and allows for easy creation of most common record types. The advanced interface is designed for system administrators with a solid DNS background, and provides layout and functionality similar to a BIND zone file allowing for the creation of nearly any record type.

Custom DNS$^{SM}$ can be used with both static and dynamic IPs and has the same automatic update capability through Custom DNS-aware clients as Dynamic DNS.

**dyndns-static -** The Static DNS service is similar to DynDNS.org's Dynamic DNS$^{SM}$ service in that it allows a host name such as yourname.dyndns.org to point to your IP address. Unlike a Dynamic DNS host, a Static DNS host does not expire after 35 days without updates, but updates take longer to propagate though the DNS system. This service is provided for up to five host names.

If your IP address does not change often or at all but you still want an easy name to remember it by (without having to purchase your own domain name), Static DNS service is ideal for you.

If you would like to use your own domain name (such as yourname.com), you need Custom DNS service which also provides full dynamic and static IP address support.

**Usage Examples**

The following example sets the dynamic-dns to **dyndns-custom** with host name **host**, user name **user**, and password **pass**:

(config)**#interface frame-relay 1.1**
(config-fr 1.16)**#dynamic-dns dyndns-custom host user pass**

# frame-relay bc *<value>*

Use the **frame-relay bc** command to set the $b_c$ (committed burst) value for a Frame Relay sublink. The value is in bits. Use the **no** form of this command to return to default.

## Syntax Description

| | |
|---|---|
| *<value>* | Specifies the committed burst value (in bits) for the sublink. |

## Default Values

By default, the committed burst value is set to 0 (no limit).

## Command History

| | |
|---|---|
| Release 4.1 | Command was introduced. |

## Functional Notes

The time interval is always one second, so this can also be considered bits per second. Shaping is performed on a sliding one-second window to make maximum use of configured bandwidth. Note that when both $b_c$ and $b_e$ are non-zero, shaping is performed on the virtual circuit. The circuit is limited to the sum of $b_c$ and $b_e$ , and it is recommended that the sum always be greater than 8000.

## Usage Examples

The following example configures the Frame Relay sublink with a committed burst value of 128,000 bits:

(config)**#interface frame-relay 1.1**
(config-fr 1.16)**#frame-relay bc 128000**

# frame-relay be *<value>*

Use the **frame-relay be** command to set the $b_e$ (excessive burst) value for a Frame Relay sublink. The value is in bits. Use the **no** form of this command to return to default.

## Syntax Description

| | |
|---|---|
| *<value>* | Specifies the excessive burst value (in bits) for the sublink. |

## Default Values

By default, the excessive burst value is set to 0 (no limit).

## Command History

| | |
|---|---|
| Release 4.1 | Command was introduced. |

## Functional Notes

The time interval is always one second, so this can also be considered bits per second. Shaping is performed on a sliding one-second window to make maximum use of configured bandwidth. Note that when both $b_c$ and $b_e$ are non-zero, shaping is performed on the virtual circuit. The circuit is limited to the sum of $b_c$ and $b_e$ , and it is recommended that the sum always be greater than 8000.

## Usage Examples

The following example configures the Frame Relay sublink with an excessive burst value of 64,000 bits:

(config)**#interface frame-relay 1.1**
(config-fr 1.16)**#frame-relay be 64000**

# frame-relay fragment *<value>*

Use the **frame-relay fragment** command to set the FRF.12 fragmentation threshold. Use the **no** form of this command to erase the configured threshold.

## Syntax Description

| | |
|---|---|
| *<value>* | Specifies the fragmentation threshold. Valid fragmentation thresholds are greater than or equal to 64 and less than or equal to 1600. |

## Default Values

No default value is necessary for this command.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Functional Notes

For Frame Relay fragmentation to take effect, rate-limiting must be enabled by setting the committed burst rate and excessive burst rate. Refer to *frame-relay bc <value>* on page 1200 and *frame-relay be <value>* on page 1201 for more information.

## Usage Examples

The following example enables FRF.12 fragmentation on a sublink:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#frame-relay bc 64000**
(config-fr 1.16)**#frame-relay be 16**
(config-fr 1.16)**#frame-relay fragmentation 100**

The following example disables FRF.12 fragmentation on a sublink:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#no frame-relay fragment**

# frame-relay interface-dlci <*value*>

Use the **frame-relay interface-dlci** command to configure the Data Link Connection Identifier (DLCI) for the Frame Relay sub-interface. This setting should match the DLCI supplied by your Frame Relay service provider. Use the **no** form of this command to remove the configured DLCI.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies numeric value of the DLCI supplied by your provider. |

## Default Values

By default, the DLCI is populated with the sub-interface identifier. For example, if configuring the virtual Frame Relay sub-interface labeled **fr 1.20**, the default DLCI is **20**.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example configures a DLCI of **72** for this Frame Relay endpoint:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#frame-relay interface-dlci 72**

# ip access-group *<name>*

Use the **ip access-group** command to create an access control list to be used for packets transmitted on or received from the specified interface. Use the **no** form of this command to disable this type of control. Variations of this command include:

**ip access-group** <*name*> **in**
**ip access-group** <*name*> **out**

## Syntax Description

| | |
|---|---|
| *<listname>* | Specifies the IP access control list name. |
| **in** | Enables access control on packets received on the specified interface. |
| **out** | Enables access control on packets transmitted on the specified interface. |

## Default Values

By default, these commands are disabled.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Functional Notes

When this command is enabled, the IP destination address of each packet must be validated before being passed through. If the packet is not acceptable per these settings, it is dropped.

## Usage Examples

The following example sets up the router to only allow Telnet traffic into the Frame Relay sub-interface:

(config)**#interface frame-relay 1.16**
(config)**#ip access-list extended TelnetOnly**
(config-ext-nacl)**#permit tcp any any eq telnet**
(config-ext-nacl)**#int frame-relay 1.16**
(config-fr 1.16)**#ip access-group TelnetOnly in**

# ip address dhcp

Use the **ip address dhcp** command to use Dynamic Host Configuration Protocol (DHCP) to obtain an address on the interface. Use the **no** form of this command to remove a configured IP address (using DHCP) and disable DHCP operation on the interface. Variables that may be used with this command to further define the DHCP configuration include:

**ip address dhcp client-id [**<*interface*> **|** <*identifier*>**] [hostname** <*"string"*>**] [track** <*name*>**]**
**[**<*administrative distance*>**]**
**ip address dhcp hostname** <*"string"*> **[no-default-route | no-domain-name | no-nameservers] [track** <*name*>**] [**<*administrative distance*>**]**
**ip address dhcp [no-default-route | no-domain-name | no-nameservers] [track** <*name*>**]**
**[**<*administrative distance*>**]**
**ip address dhcp track** <*name*> **[**<*administrative distance*>**]**

## Syntax Description

| | |
|---|---|
| <*administrative distance*> | Optional. Specifies the administrative distance to use when adding the DHCP gateway into the route table. It is used to determine the best route when multiple routes to the same destination exist. The smaller the administrative distance the more reliable the route. Range is 1 to 255. |
| **client-id** | Optional. Specifies the client identifier used when obtaining an IP address from a DHCP server. |
| <*interface*> | Specifies an interface, thus defining the client identifier as the hexadecimal MAC address of the specified interface (including a hexadecimal number added to the front of the MAC address to identify the media type).<br>For example, specifying the **client-id ethernet 0/1** (where the Ethernet interface has a MAC address of d217.0491.1150) defines the client identifier as **01:d2:17:04:91:11:50** (where 01 defines the media type as Ethernet). Refer to *hardware-address* <*mac address*> on page 1973 for a detailed listing of media types. |
| <*identifier*> | Specifies a custom client-identifier using a text string (that is converted to a hexadecimal equivalent) or 7 to 28 hexadecimal numbers (with colon delimiters).<br>For example, a custom client identifier of **0f:ff:ff:ff:ff:51:04:99:a1** may be entered using the <*identifier*> option. |
| **hostname** <*"string"*> | Optional. Specifies a text string (to override the global router name) to use as the name in the DHCP option 12 field. The string is enclosed in quotation marks and can consist of up to 35 characters. |
| **no-default-route** | Optional. Specifies that no default route is obtained via DHCP. |
| **no-domain-name** | Optional. Specifies that no domain name is obtained via DHCP. |
| **no-nameservers** | Optional. Specifies that no DNS servers are obtained via DHCP. |
| **track** <*name*> | Optional. Attaches a network monitoring track to the DHCP client. The DHCP gateway route for this client will only reside in the route table while the track is in the pass state. For more information on configuring track objects, refer to *track* <*name*> on page 691. |

## Default Values

| | |
|---|---|
| *<administrative distance>* | By default, the administrative distance value is 1. |
| **client-id** | Optional. By default, the client identifier is populated using the following formula:<br>TYPE: INTERFACE SPECIFIC INFO : MAC ADDRESS<br>Where TYPE specifies the media type in the form of one hexadecimal byte (refer to *hardware-address <mac address>* on page 1973 for a detailed listing of media types), and the MAC ADDRESS is the Media Access Control (MAC) address assigned to the first Ethernet interface in the unit in the form of six hexadecimal bytes. (For units with a single Ethernet interface, the MAC ADDRESS assigned to Ethernet 0/1 is used in this field).<br>INTERFACE SPECIFIC INFO is only used for Frame Relay interfaces and can be determined using the following:<br>FR_PORT#: Q.922 ADDRESS<br>Where the FR_PORT# specifies the label assigned to the virtual Frame Relay interface using four hexadecimal bytes. For example, a virtual Frame Relay interface labeled 1 would have a FR_PORT# of 00:00:00:01.<br>The Q.922 ADDRESS field is populated using the following: |

| 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
|---|---|---|---|---|---|---|---|
| DLCI (high order) | | | | | | C/R | EA |
| DLCI (lower) | | | | FECN | BECN | DE | EA |

Where the FECN, BECN, C/R, DE, and high order extended address (EA) bits are assumed to be 0 and the lower order EA bit is set to 1.

The following list provides a few example DLCIs and associated Q.922 address:

DLCI (decimal) / Q.922 address (hex)

16 / 0x0401
50 / 0x0C21
60 / 0x0CC1
70 / 0x1061
80 / 0x1401

| | |
|---|---|
| **hostname** <*"string"*> | By default, the host name is the name configured using the Global Configuration **hostname** command. |

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |
| Release 8.1 | Command was expanded to include ATM sub-interface. |
| Release 13.1 | Command was expanded to include track and administrative distance. |

## Functional Notes

DHCP allows interfaces to acquire a dynamically assigned IP address from a configured DHCP server on the network. Many Internet Service Providers (ISPs) require the use of DHCP when connecting to their services. Using DHCP reduces the number of dedicated IP addresses the ISP must obtain. Consult your ISP to determine the proper values for the **client-id** and **hostname** fields.

## Usage Examples

The following example enables DHCP operation on the virtual Frame Relay interface (labeled 1.16):

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#ip address dhcp**

The following example enables DHCP operation on the virtual Frame Relay interface (labeled 1.16) utilizing hostname **adtran** and does not allow obtaining a default route, domain name, or nameservers. It also sets the administrative distance as **5**:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#ip address dhcp hostname “adtran” no-default-route no-domain-name no-nameservers 5**

# ip address *<ip address> <subnet mask>*

Use the **ip address** command to define an IP address on the specified interface. Use the optional **secondary** keyword to define a secondary IP address. Use the **no** form of this command to remove a configured IP address. Variations of this command include:

**ip address** *<ip address> <subnet mask>*
**ip address** *<ip address> <subnet mask>* **secondary**

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies a valid IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| *<subnet mask>* | Specifies the subnet mask that corresponds to a range of IP addresses (network) or a specific host. Subnet masks can be expressed in dotted decimal notation (for example, **255.255.255.0**) or as a prefix length (for example, **/24**). |
| **secondary** | Optional. Configures a secondary IP address for the specified interface. |

## Default Values

By default, there are no assigned IP addresses.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

Use secondary IP addresses to allow dual subnets on a single interface (when you need more IP addresses than the primary subnet can provide). When using secondary IP addresses, avoid routing loops by verifying that all devices on the network segment are configured with secondary IP addresses on the secondary subnet.

## Usage Examples

The following example configures a secondary IP address of **192.22.72.101/30**:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#ip address 192.22.72.101 255.255.255.252 secondary**

# ip dhcp

Use the **ip dhcp** command to release or renew the DHCP IP address. This command is only applicable when using DHCP for IP address assignment. Variations of this command include:

**ip dhcp release**
**ip dhcp renew**

## Syntax Description

| | |
|---|---|
| **release** | Releases DHCP IP address. |
| **renew** | Renews DHCP IP address. |

## Default Values

No default values required for this command.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Usage Examples

The following example releases the IP DHCP address for the virtual interface:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#ip dhcp release**

# ip ffe

Use the **ip ffe** command to enable the FastFlow Engine (FFE) on this interface with the default number of entries. Use the **no** form of these commands to disable the FFE. Variations of this command include:

**ip ffe max-entries** <*entries*>

*Issuing this command will cause all FFE entries on this interface to be cleared.*

## Syntax Description

| | |
|---|---|
| **max-entries** <*entries*> | Specifies the maximum number of entries stored in the flow table. Valid range is from 1 to 8192. |

## Default Values

By default, the FFE is disabled. The default number of **max-entries** is 4096.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Functional Notes

The FFE can be used to help reduce routing overhead and thus reduce overall routing times. Routing times are reduced by the creation of a flow table on the ingress interface. The maximum number of entries that can be stored in the flow table at any one time may be specified by using the **max-entries** subcommand. When features such as firewall, VPN, and static filters are enabled, routing time reduction may not be realized.

## Usage Examples

The following example enables the FFE and sets the maximum number of entries in the flow table to 50:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#ip ffe max-entries 50**

## Technology Review

The FastFlow system goal is to increase IP packet throughput by moving as much of the packet processing into the engine as possible. Packets are classified into flows based upon the IP protocol (TCP, UDP, ICMP, etc.), the source and destination IP addresses, IP TOS, and the protocol-specific information such as the source and destination port numbers. Flows are defined as the unidirectional representation of a conversation between two IP hosts. Each ingress interface keeps its own flow table, a collection of flow entries.

The first packet in a flow that is forwarded through the unit will build a flow entry. When a flow entry is looked up but no entry is found, a FastFlowBuilder object is allocated and attached to the packet. As the packet passes through the various processing layers, each subsystem will add processing to the FastFlowBuilder. When packet is about to be forwarded out of the egress interface, the FastFlowBuilder will be finalized. That is, the flow entry being built will be checked for completeness and committed to the flow table on the ingress interface. Subsequent flow matches can then bypass the normal processing layers.

# ip helper-address *<ip address>*

Use the **ip helper-address** command to configure the AOS to forward User Datagram Protocol (UDP) broadcast packets received on the interface. Use the **no** form of this command to disable forwarding packets.

> NOTE *The **ip helper** command must be used in conjunction with the **ip forward-protocol** command to configure the AOS to forward UDP broadcast packets. Refer to ip forward-protocol udp <value> on page 532 for more information.*

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies the destination IP address for the forwarded UDP packets. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

## Default Values

By default, broadcast UDP packets are not forwarded.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

When used in conjunction with the **ip forward-protocol** command, the **ip helper-address** feature allows you to customize which broadcast packets are forwarded.

To implement the helper address feature, assign a helper-address(es) (specifying the device that needs to receive the broadcast traffic) to the interface closest to the host that transmits the broadcast packets. When broadcast packets (of the specified type forwarded using the **ip forward-protocol** command) are received on the interface, they will be forwarded to the device that needs the information.

Only packets meeting the following criteria are considered eligible by the **ip helper-address** feature:

1. The packet IP protocol is User Datagram Protocol (UDP).
2. Any UDP port specified using the **ip forward-protocol** command.
3. The media access control (MAC) address of the frame is an all-ones broadcast address (ffff.ffff.ffff).
4. The destination IP address is broadcast defined by all ones (255.255.255.255) or a subnet broadcast (for example, 192.33.4.251 for the 192.33.4.248/30 subnet).

## Usage Examples

The following example forwards all DNS broadcast traffic to the DNS server with IP address **192.33.5.99**:

(config)**#interface frame-relay 1.16**
(config)**#ip forward-protocol udp domain**
(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#ip helper-address 192.33.5.99**

# ip igmp

Use the **ip igmp** command to configure multicasting-related functions for the interface. Variations of this command include:

**ip igmp immediate-leave**
**ip igmp last-member-query-interval** <*milliseconds*>
**ip igmp querier-timeout** <*seconds*>
**ip igmp query-interval** <*seconds*>
**ip igmp query-max-response-time** <*seconds*>
**ip igmp static-group** <*address*>
**ip igmp version [1 | 2]**

## Syntax Description

| | |
|---|---|
| **immediate-leave** | Specifies that if only one host (or IGMP snooping switch) is connected to the interface, when a leave is received, multicast of that group is immediately terminated as opposed to sending a group query and timing out the group if no device responds. Works in conjunction with **ip igmp last-member-query-interval**. Applies to all groups when configured. Use the **no** form of this command to disable the **immediate-leave** feature. |
| **last-member-query-interval** <*milliseconds*> | Controls the timeout (in milliseconds) used to detect whether any group receivers remain on an interface after a receiver leaves a group. If a receiver sends a leave-group message (IGMP Version 2), the router sends a group-specific query on that interface. After twice the time specified by this command plus as much as one second longer, if no receiver responds, the router removes that interface from the group and stops sending that group's multicast packets to the interface. Range is 100 to 65,535 ms. Use the **no** form of this command to return to the default setting. |
| **querier-timeout** <*seconds*> | Specifies the interval (in seconds) that the router waits after the current querier's last query before it takes over as querier (IGMP V2). Range is 60 to 300 seconds. Use the **no** form of this command to return to the default setting. |
| **query-interval** <*seconds*> | Specifies the interval (in seconds) at which IGMP queries are sent on an interface. Host query messages are addressed to the all-hosts multicast group with an IP TTL of 1. The router uses queries to detect whether multicast group members are on the interface and to select an IGMP designated router for the attached segment (if more than one multicast router exists). Only the designated router for the segment sends queries. For IGMP V2, the designated router is the router with the lowest IP address on the segment. Range is 0 to 65,535 seconds. Use the **no** form of this command to return to the default setting. |
| **query-max-response-time** <*seconds*> | Specifies the maximum response time (in seconds) advertised by this interface in queries when using IGMP V2. Hosts are allowed a random time within this period to respond, reducing response bursts. Use the **no** form of this command to return to the default setting. |

## Syntax Description

| | |
|---|---|
| **static-group** <*address*> | Configures the router's interface to be a statically-connected member of the specified group. Packets received on the correct RPF interface are forwarded to this interface regardless of whether any receivers have joined the specified group using IGMP. Use the **no** form of this command to remove a configured static group. |
| **version [1 \| 2]** | Sets the interface's IGMP version. Use the **no** form of this command to return to the default setting. |

## Default Values

| | |
|---|---|
| **ip igmp immediate-leave** | No default |
| **ip igmp last-member-query-interval** | 1000 milliseconds |
| **ip igmp querier-timeout** | 2x the **query-interval** value |
| **ip igmp query-interval** | 60 seconds |
| **ip igmp query-max-response-time** | 10 seconds |
| **ip igmp static-group** | No default |
| **ip igmp version** | Version 1 |

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |

## Usage Examples

The following example sets the query message interval on the interface to **200** milliseconds:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#ip igmp last-member-query-interval 200**

# ip mcast-stub downstream

Use the **ip mcast-stub downstream** command to enable multicast forwarding and IGMP (router mode) on an interface and place it in multicast stub downstream mode. Use the **no** form of this command to disable.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |

## Functional Notes

This command is used in IP multicast stub applications in conjunction with the **ip mcast-stub helper-address** and **ip mcast-stub upstream** commands. Downstream interfaces connect to segments with multicast hosts. Multiple interfaces may be configured in downstream mode; however, interfaces connecting to the multicast network (upstream) should not be configured in downstream mode. Interfaces configured as downstream should have the lowest IP address of all IGMP-capable routers on the connected segment in order to be selected as the designated router and ensure proper forwarding. Refer to *ip mcast-stub helper-address <ip address>* on page 546 for more information.

## Usage Examples

The following example enables multicast forwarding and IGMP on the interface:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#ip mcast-stub downstream**

# ip mcast-stub fixed

Use the **ip mcast-stub fixed** command to allow forwarding of multicast traffic on a selected interface after enabling multicast routing. Use the **no** form of this command to disable this mode.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Functional Notes

Multicast routing must be enabled prior to setting **ip mcast-stub fixed** on the selected interface. Also, use the **ip igmp static-group** <*A.B.C.D*> command to receive multicast traffic without host-initiated Internet Group Management Protocol (IGMP) activity on the selected interface. Otherwise, all host-initiated IGMP transactions will enter multicast routes on the router's interface involved with IGMP activities.

## Usage Examples

The following example enables multicast traffic forwarding and IGMP on the interface:

(config)**#interface frame-relay 1**
(config-fr 1.16)**#ip mcast-stub fixed**

# ip mcast-stub helper-enable

Use the **ip mcast-stub helper-enable** command to assign the **ip mcast-stub helper-address** as the IGMP proxy. Use the **no** form of this command to disable.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |
| Release 10.1 | Command was expanded to include Frame Relay sub-interfaces. |

## Functional Notes

This command is used in IP multicast stub applications in conjunction with the **ip mcast-stub helper-address**, **ip mcast-stub upstream**, and **ip mcast-stub downstream** commands. When enabled, the interface becomes a helper forwarding interface. The IGMP host function is dynamically enabled and the interface becomes the active upstream interface, enabling the unit to perform as an IGMP proxy. Refer to *ip mcast-stub helper-address <ip address>* on page 546, *ip mcast-stub downstream* on page 1216, and *ip mcast-stub upstream* on page 1219 for more information.

## Usage Examples

The following example sets the helper address as the IGMP proxy:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#ip mcast-stub helper-enable**

# ip mcast-stub upstream

Use the **ip mcast-stub upstream** command to enable multicast forwarding on an interface and place it in multicast stub upstream mode. Use the **no** form of this command to disable.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |

## Functional Notes

This command is used in IP multicast stub applications in conjunction with the **ip mcast-stub helper-address** and **ip mcast-stub downstream** commands. When enabled, the interface becomes a candidate to be a helper forwarding interface. If chosen as the best path toward the helper address by the router's unicast route table, the IGMP host function is dynamically enabled and the interface becomes the active upstream interface, enabling the router to perform as an IGMP proxy. Though multiple interfaces may be candidates, no more than one interface will actively serve as the helper forwarding interface. Refer to *ip mcast-stub helper-address <ip address>* on page 546 and *ip mcast-stub downstream* on page 1216 for more information.

## Usage Examples

The following example enables multicast forwarding on the interface:

```
(config)#interface frame-relay 1.16
(config-fr 1.16)#ip mcast-stub upstream
```

# ip ospf

Use the **ip ospf** command to customize OSPF settings (if needed). Use the **no** form of these commands to return to the default settings. Variations of this command include:

**ip ospf authentication-key** <*password*>
**ip ospf cost** <*value*>
**ip ospf dead-interval** <*seconds*>
**ip ospf hello-interval** <*seconds*>
**ip ospf message-digest-key [1 | 2] md5** <*key*>
**ip ospf priority** <*value*>
**ip ospf retransmit-interval** <*seconds*>
**ip ospf transmit-delay** <*seconds*>

## Syntax Description

| | |
|---|---|
| **authentication-key** <*password*> | Assigns a simple-text authentication password to be used by other routers using the OSPF simple password authentication. |
| **cost** <*value*> | Specifies the OSPF cost of sending a packet on the interface. This value overrides any computed cost value. Range is 1 to 65,535. |
| **dead-interval** <*seconds*> | Sets the maximum interval (in seconds) allowed between hello packets. If the maximum is exceeded, neighboring devices will determine that the device is down. Range is 0 to 32,767 seconds. |
| **hello-interval** <*seconds*> | Specifies the interval (in seconds) between hello packets sent on the interface. Range is 0 to 32,767 seconds. |
| **message-digest-key [1 \| 2] md5** <*key*> | Configures OSPF Message Digest 5 (MD5) authentication (16-byte max) keys. |
| **priority** <*value*> | Sets the OSPF priority. The value set in this field helps determine the designated router for this network. Range is 0 to 255. |
| **retransmit-interval** <*seconds*> | Specifies the interval (in seconds) between link-state advertisements (LSAs). Range is 0 to 32,767 seconds. |
| **transmit-delay** <*seconds*> | Sets the estimated time (in seconds) required to send an LSA on the interface. Range is 0 to 32,767 seconds. |

## Default Values

| | |
|---|---|
| **dead-interval** <*seconds*> | 40 seconds |
| **hello-interval** <*seconds*> | 10 seconds: Ethernet, point-to-point, Frame Relay, Tunnel, and PPP |
| **retransmit-interval** <*seconds*> | 5 seconds |
| **transmit-delay** <*seconds*> | 1 second |

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Usage Examples

The following example sets the maximum number of seconds allowed between hello packets to 25,000:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#ip ospf dead-interval 25000**

# ip ospf authentication

Use the **ip ospf authentication** command to authenticate an interface that is performing OSPF authentication. Use the **no** form of this command to return to the default setting. Variations of this command include:

**ip ospf authentication**
**ip ospf authentication message-digest**
**ip ospf authentication null**

## Syntax Description

| | |
|---|---|
| **message-digest** | Optional. Selects message-digest authentication type. |
| **null** | Optional. Specifies that no authentication be used. |

## Default Values

By default, this is set to null (meaning no authentication is used).

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Usage Examples

The following example specifies that no authentication will be used on the Frame Relay interface:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#ip ospf authentication null**

# ip ospf network

Use the **ip ospf network** command to specify the type of network on this interface. Use the **no** form of this command to return to the default setting. Variations of this command include:

**ip ospf network broadcast**
**ip ospf network point-to-point**

## Syntax Description

| | |
|---|---|
| **broadcast** | Sets the network type for broadcast. |
| **point-to-point** | Sets the network type for point-to-point. |

## Default Values

By default, Ethernet defaults to broadcast. PPP and Frame Relay default to point-to-point.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Functional Notes

A point-to-point network will not elect designated routers.

## Usage Examples

The following example designates a broadcast network type:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#ip ospf network broadcast**

# ip pim sparse-mode

Use the **ip pim sparse-mode** command to enable protocol-independent multicast (PIM) sparse mode for this interface. Use the **no** form of this command to disable PIM sparse mode.

## Syntax Description

No subcommands.

## Default Values

By default, PIM sparse mode for this interface is disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Functional Notes

PIM Sparse Mode is a multicast routing protocol that makes use of the unicast forwarding table. It builds unidirectional shared trees rooted at a Rendezvous Point (RP) for a multicast group or a shortest-path tree rooted at a specific source for a multicast group.

## Usage Examples

The following example enables PIM sparse mode on the interface:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#ip pim sparse-mode**

# ip pim-sparse dr-priority <*value*>

Use the **ip pim-sparse dr-priority** command to specify the priority for the designated router (DR). This command modifies the routers priority in the DR election process. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the priority of this interface (to be used when determining the DR). Valid range is 1 to 4,294,967,295. |

## Default Values

By default, the priority of all PIM interfaces is 1.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Functional Notes

Interfaces advertise their configured priority values in the hello messages transmitted on the interface. Routers use the priority values to determine the appropriate DR. The router on the network segment with the highest priority is selected as the DR. If a hello message is received on the interface from a router on the network segment and it does not contain a priority, the entire network segment defaults to DR selection based on IP addresses instead of priority. In this instance, the DR is selected as the router on the network segment that has the highest IP address. AOS will always include a priority in all transmitted hello messages. If no priority is specifically designated by the user, the priority is set as the default of 1.

## Usage Examples

The following specifies a priority of **100** on the Frame Relay sub-interface:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#ip pim-sparse dr-priority 100**

# ip pim-sparse hello-timer *<value>*

Use the **ip pim-sparse hello-timer** command to specify protocol-independent multicast (PIM) sparse hello timer period. This is the time interval at which periodic hellos are sent out on all interfaces of a PIM-capable router. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| *<value>* | Specifies the interval (in seconds) at which periodic hellos are sent out of the interface. Valid range is 10 to 3600 seconds. |

## Default Values

By default, the hellos are transmitted on PIM interfaces every 60 seconds.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Functional Notes

Hello messages are used to inform neighbors of a router's presence. Hello messages normally generate a small amount of traffic on an interface. Setting the **hello-timer** to a small interval increases the number of hellos sent (thus increasing the amount of traffic). Set the **hello-timer** to a reasonable value, taking into consideration the bandwidth available on the interface.

## Usage Examples

The following example specifies hellos be sent on the Frame Relay sub-interface every **3600** seconds:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#ip pim-sparse hello-timer 3600**

# ip pim-sparse nbr-timeout <*value*>

Use the **ip pim-sparse nbr-timeout** command to specify protocol-independent multicast (PIM) sparse neighbor timeout. This is the time interval after which a PIM-capable router will consider a neighbor not present. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the time interval in seconds after which a neighbor is considered not present. Valid range is 30 to 10,800 seconds. |

## Default Values

By default, the nbr-timeout is set to 105 seconds.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example sets the nbr-timeout to **300** seconds:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#ip pim-sparse nbr-timeout 300**

# ip pim-sparse override-interval <*value*>

Use the **ip pim-sparse override-interval** command to specify the protocol-independent multicast (PIM) sparse join/prune override interval. This delay interval is the period after a join/prune that another router on the LAN may override a join/prune. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the delay time in milliseconds. Valid range is 0 to 65,535 milliseconds. |

## Default Values

By default, the override interval is set to 2500 milliseconds.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example sets the override interval to **3000** milliseconds:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#ip pim-sparse override-interval 3000**

# ip pim-sparse propagation-delay <*value*>

Use the **ip pim-sparse propagation-delay** command to specify the expected propagation delay for join/prune messages. Set the propagation delay (in milliseconds) to estimate the amount of delay found in the local link. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the expected propagation delay in the local link in milliseconds. Valid range is 0 to 32,767 milliseconds. |

## Default Values

By default, the propagation delay is set to 500 milliseconds.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example sets the propagation delay to **300** milliseconds on the Frame Relay sub-interface:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#ip pim-sparse propagation-delay 300**

# ip policy route-map *<name>*

Use the **ip policy route-map** command to assign a policy route-map to this interface. Use the **no** form of this command to remove the route-map policy.

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies the name of the policy route map to assign to this interface. |

## Default Values

By default, no policy route map is assigned to this interface.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example assigns the policy route map **policy1** to the interface:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#ip policy route-map policy1**

# ip proxy-arp

Use the **ip proxy-arp** to enable proxy Address Resolution Protocol (ARP) on the interface. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, proxy ARP is enabled.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

In general, the principle of proxy ARP allows a router to insert its IP address in the source IP address field of a packet (if the packet is from a host on one of its subnetworks). This allows hosts to reach devices on other subnetworks without implementing routing or specifying a default gateway.

If proxy ARP is enabled, the AOS will respond to all proxy ARP requests with its specified MAC address and forward packets accordingly.

Enabling proxy ARP on an interface may introduce unnecessary ARP traffic on the network.

## Usage Examples

The following enables proxy ARP on the Frame Relay sub-interface:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#ip proxy-arp**

# ip rip receive version

Use the **ip rip receive version** command to configure the RIP version the unit accepts in all RIP packets received on the interface. Use the **no** form of this command to restore the default value. Variations of this command include:

**ip rip receive version 1**
**ip rip receive version 2**

## Syntax Description

| | |
|---|---|
| **1** | Accepts only received RIP version 1 packets on the interface. |
| **2** | Accepts only received RIP version 2 packets on the interface. |

## Default Values

By default, all interfaces implement RIP version 1 (the default value for the **version** command).

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

Use the **ip rip receive version** command to specify a RIP version that will override the **version** (in the Router RIP) configuration.

The AOS only accepts one version (either 1 or 2) on a given interface.

## Usage Examples

The following example configures a Frame Relay sub-interface to accept only RIP version 2 packets:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#ip rip receive version 2**

# ip rip send version

Use the **ip rip send version** command to configure the RIP version the unit sends in all RIP packets transmitted on the interface. Use the **no** form of this command to restore the default value. Variations of this command include:

**ip rip send version 1**
**ip rip send version 2**

## Syntax Description

| | |
|---|---|
| **1** | Transmits only RIP version 1 packets on the interface. |
| **2** | Transmits only RIP version 2 packets on the interface. |

## Default Values

By default, all interfaces transmit RIP version 1 (the default value for the **version** command).

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

Use the **ip rip send version** to specify a RIP version that will override the **version** (in the Router RIP) configuration.

The AOS only transmits one version (either 1 or 2) on a given interface.

## Usage Examples

The following example configures a Frame Relay sub-interface to transmit only RIP version 2 packets:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#ip rip send version 2**

# ip rip summary-address *<ip address> <subnet mask>*

Use the **ip rip summary-address** command to manually summarize the routes Routing Information Protocol (RIP) will advertise and send out a specified interface. Use the **no** form of this command to disable this mode.

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies the summarized network IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.0**). |
| *<subnet mask>* | Specifies the subnet mask that corresponds to the range of IP addresses (network). Subnet masks can be expressed in dotted decimal notation (for example, **255.255.255.0**) or as a prefix length (for example, **/24**). |

## Default Values

By default, no manual summarization is applied by RIP.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Functional Notes

Unlike the automatic summarization on classful network boundaries, only specific network advertisements are made by RIP using the **ip rip summary-address** command. This command is only effective if RIP version 2 is configured.

## Usage Examples

The following example enables manual summarization on the specified IP address:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#ip rip summary-address 10.10.123.0 255.255.255.0**

# ip route-cache

Use the **ip route-cache** command to enable fast-cache switching on the interface. Use the **no** form of this command to disable fast-cache switching and return to process switching mode.

*Using Network Address Translation (NAT) or the AOS firewall capabilities on an interface requires process switching mode (using the **no ip route-cache** command).*

## Syntax Description

No subcommands.

## Default Values

By default, fast-cache switching is enabled on all Ethernet and virtual Frame Relay sub-interfaces. IP route cache is enabled for all virtual PPP interfaces.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |

## Functional Notes

Fast switching allows an IP interface to provide optimum performance when processing IP traffic.

## Usage Examples

The following example enables fast-cache switching on a Frame Relay sub-interface:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#ip route-cache**

# ip unnumbered *<interface>*

Use the **ip unnumbered** command to use the IP address assigned to the specified interface for all IP processing on the active interface. Use the **no** form of this command to remove the unnumbered configuration.

## Syntax Description

| | |
|---|---|
| *<interface>* | Specifies the interface that contains the IP address to use as the source address for all packets transmitted on this interface. Specify an interface in the format *<interface type [slot/port \| slot/port.sub-interface id \| interface id \| interface id.sub-interface id]>*. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **show ip unnumbered ?** for a list of valid interfaces. |

## Default Values

By default, all interfaces are configured to use a specified IP address (using the **ip address** command).

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 11.1 | Command was expanded to include demand interfaces. |

## Functional Notes

If **ip unnumbered** is enabled on an interface, all IP traffic from the interface will use a source IP address taken from the specified interface. For example, specifying **ip unnumbered eth 0/1** while in the Frame Relay Sub-Interface Configuration mode configures the Frame Relay sub-interface to use the IP address assigned to the Ethernet interface for all IP processing. In addition, the AOS uses the specified interface information when sending route updates over the unnumbered interface.

## Usage Examples

The following example configures the Frame Relay interface (labeled **frame-relay 1.16**) to use the IP address assigned to the Ethernet interface (**eth 0/1**):

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#ip unnumbered eth 0/1**

# ip urlfilter *<name>*

Use the **ip urlfilter** command to apply a URL filter to the interface for all inbound or outbound traffic. Use the **no** form of this command to remove the URL filter from an interface. Variations of this command include:

**ip urlfilter** *<name>* **in**
**ip urlfilter** *<name>* **out**

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies the URL filter name to use on the interface. |
| **in** | Applies the filter to the inbound traffic. |
| **out** | Applies the filter to the outbound traffic. |

## Default Values

By default, there are no URL filters applied to any interfaces.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Functional Notes

The firewall must be enabled using the **ip firewall** command in order to use URL filters. The URL filter must be created by using the **ip urlfilter** *<name>* **http** command before applying it to the interface. See *ip urlfilter <name> http* on page 598 for more information on using this command.

## Usage Example

The following example performs URL filtering on all traffic entering through a Frame Relay sub-interface and matches the URL filter named **MyFilter**:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#ip urlfilter MyFilter in**

# lldp receive

Use the **lldp receive** command to allow LLDP packets to be received on this interface. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, all interfaces are configured to send and receive LLDP packets.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Usage Examples

The following example configures the Frame Relay sub-interface to receive LLDP packets:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#lldp receive**

# lldp send

Use the **lldp send** command to configure this interface to transmit LLDP packets or to control the types of information contained in the LLDP packets transmitted by this interface. Use the **no** form of these commands to disable these features. Variations of this command include:

**lldp send**
**lldp send management-address**
**lldp send port-description**
**lldp send system-capabilities**
**lldp send system-description**
**lldp send system-name**
**lldp send-and-receive**

## Syntax Description

| | |
|---|---|
| **management-address** | Enables transmission of management address information on this interface. |
| **port-description** | Enables transmission of port description information on this interface. |
| **system-capabilities** | Enables transmission of this device's system capabilities on this interface. |
| **system-description** | Enables transmission of this device's system description on this interface. |
| **system-name** | Enables transmission of this device's system name on this interface. |
| **and-receive** | Configures this interface to both transmit and receive LLDP packets. |

## Default Values

Be default, all interfaces are configured to transmit and receive LLDP packets of all types.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Functional Notes

Individual LLDP information can be enabled or disabled using the various forms of the **lldp send** command. For example, use the **lldp send and-receive** command to enable transmit and receive of all LLDP information. Then use the **no lldp send port-description** command to prevent LLDP from transmitting port description information.

## Usage Examples

The following example configures the Frame Relay sub-interface to transmit LLDP packets containing all enabled information types:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#lldp send**

The following example configures the Frame Relay sub-interface to transmit and receive LLDP packets containing all information types:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#lldp send and-receive**

# media-gateway ip

Use the **media-gateway ip** command to associate an IP address source to use for RTP traffic. When configuring VoIP, RTP traffic needs an IP address to be associated with it. However, some interfaces allow "dynamic" configuration of IP addresses, and thus, this value could change periodically. Use the **no** form of these commands to disable these functions.Variations of this command include:

**media-gateway ip loopback** <*ip address*>
**media-gateway ip primary**
**media-gateway ip secondary** <*ip address*>

## Syntax Description

| | |
|---|---|
| **loopback** <*ip address*> | Use an IP address statically defined to a loopback interface. Helpful when using a single IP address across multiple WAN interfaces for RTP traffic. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| **primary** | Use the IP address that is configured as primary on this interface. Applies to static, DHCP, or negotiated addresses. |
| **secondary** <*ip address*> | Use the statically defined secondary IP address of this interface to be used for RTP traffic. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

## Default Values

By default, **media-gateway ip** is disabled.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example configures the unit to use the primary IP address for RTP traffic:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#media-gateway ip primary**

# mtu <*size*>

Use the **mtu** command to configure the maximum transmit unit (MTU) size for the active interface. Use the **no** form of this command to return to the default value.

## Syntax Description

| | | |
|---|---|---|
| <*size*> | Configures the window size for transmitted packets. The valid ranges for the various interfaces are listed below: | |
| | ATM interfaces | 64 to 1520 |
| | Demand interfaces | 64 to 1520 |
| | Ethernet interfaces | 64 to 1500 |
| | FDL interfaces | 64 to 256 |
| | HDLC interfaces | 64 to 1520 |
| | Loopback interfaces | 64 to 1500 |
| | Tunnel interfaces | 64 to 18,190 |
| | Virtual Frame Relay sub-interfaces | 64 to 1520 |
| | Virtual PPP interfaces | 64 to 1500 |

## Default Values

| | | |
|---|---|---|
| <*size*> | The default values for the various interfaces are listed below: | |
| | ATM interfaces | 1500 |
| | Demand interfaces | 1500 |
| | Ethernet interfaces | 1500 |
| | FDL interfaces | 256 |
| | HDLC interfaces | 1500 |
| | Loopback interfaces | 1500 |
| | Tunnel interfaces | 1500 |
| | Virtual Frame Relay sub-interfaces | 1500 |
| | Virtual PPP interfaces | 1500 |

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

OSPF will not become adjacent on links where the MTU sizes do not match. If router A and router B are exchanging hello packets but their MTU sizes do not match, they will never reach adjacency. This is by design and required by the RFC.

## Usage Examples

The following example specifies an MTU of 1200 on the Frame Relay interface:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#mtu 1200**

# spanning-tree bpdufilter

Use the **spanning-tree bpdufilter** command to block BPDUs from being transmitted and received on this interface. To return to the default value, use the **no** form of this command. Variations of this command include:

**spanning-tree bpdufilter disable**
**spanning-tree bpdufilter enable**

## Syntax Description

| | |
|---|---|
| **disable** | Disables the BPDU filter. |
| **enable** | Enables the BPDU filter. |

## Default Values

By default, this command is set to disable.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

The purpose of this command is to remove a port from participation in the spanning tree. This might be beneficial while debugging a network setup. It normally should not be used in a live network.

## Usage Examples

The following example enables the BPDU filter on the interface:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#spanning-tree bpdufilter enable**

# spanning-tree bpduguard

Use the **spanning-tree bpduguard** command to block BPDUs from being received on this interface. To return to the default value, use the **no** form of this command. Variations of this command include:

**spanning-tree bpduguard disable**
**spanning-tree bpduguard enable**

## Syntax Description

| | |
|---|---|
| **disable** | Disables the BPDU block. |
| **enable** | Enables the BPDU block. |

## Default Values

By default, this command is set to disable.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following example enables the BPDU guard on the interface:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#spanning-tree bpduguard enable**

# spanning-tree edgeport

Use the **spanning-tree edgeport** command to set this interface to be an edgeport. This command overrides the Global setting (refer to *spanning-tree edgeport default* on page 678). Use the **no** form of this command to return to the default value.

## Syntax Description

No subcommands.

## Default Values

By default, this command is set to disable.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following example configures the interface to be an edgeport:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#spanning-tree edgeport**

An individual interface can be configured to not be considered an edgeport. For example:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#spanning-tree edgeport disable**

or

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#no spanning-tree edgeport**

# spanning-tree link-type

Use the **spanning-tree link-type** command to configure the spanning-tree protocol link type for an interface. To return to the default value, use the **no** form of this command. Variations of this command include:

**spanning-tree link-type auto**
**spanning-tree link-type point-to-point**
**spanning-tree link-type shared**

## Syntax Description

| | |
|---|---|
| **auto** | Determines link type by the port's duplex settings. |
| **point-to-point** | Manually sets link type to point-to-point, regardless of duplex settings. |
| **shared** | Manually sets link type to shared, regardless of duplex settings. |

## Default Values

By default, a port is set to auto.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

This command overrides the default link-type setting determined by the duplex of the individual port. By default a port configured for half-duplex is set to **shared** link type, and a port configured for full-duplex is set to **point-to-point** link type. Setting the link type manually overrides the default and forces the port to use the specified link type. Using the **link-type auto** command, restores the convention of determining link-type based on duplex settings.

## Usage Examples

The following example forces the link type to point-to-point, even if the port is configured to be half-duplex:

(config)**#bridge 1 protocol ieee**
(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#spanning-tree link-type point-to-point**

## Technology Review

Rapid transitions are possible in rapid spanning-tree protocol (RSTP) by taking advantage of point-to-point links (a port is connected to exactly one other bridge) and edge-port connections (a port is not connected to any additional bridges). Setting the link type to **auto** allows the spanning tree to automatically configure the link type based on the duplex of the link. Setting the link type to **point-to-point** allows a half-duplex link to act as if it were a point-to-point link.

# spanning-tree path-cost *<value>*

Use the **spanning tree path-cost** command to assign a cost to a bridge group that is used when computing the spanning-tree root path. To return to the default path-cost value, use the **no** form of this command.

## Syntax Description

| | |
|---|---|
| *<value>* | Assigns a path cost value for spanning calculations to the bridge interface. Valid range is 0 to 65,535. |

## Default Values

Be default, the path-cost value is set at 19.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

The specified value is inversely proportional to the likelihood the bridge interface will be chosen as the root path. Set the path-cost value lower to increase the chance the interface will be the root. To obtain the most accurate spanning-tree calculations, develop a system for determining path costs for links and apply it to all bridged interfaces.

## Usage Examples

The following example assigns a path cost of 100 on a Frame Relay sub-interface:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#spanning-tree path-cost 100**

## Technology Review

Spanning-tree protocol provides a way to prevent loopback or parallel paths in bridged networks. Using the priority values and path costs assigned to each bridging interface, the spanning-tree protocol determines the root path and identifies whether to block or allow other paths.

# spanning-tree port-priority *<value>*

Use the **spanning-tree priority** command to select the priority level of a port associated with a bridge. To return to the default bridge-group priority value, use the **no** version of this command.

## Syntax Description

| | |
|---|---|
| *<value>* | Priority value for the bridge group; the lower the value, the higher the priority. Valid range is 0 to 255. |

## Default Values

Be default, the bridge-group priority value is set at 28.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

The only time that this priority level is used is when two interfaces with a path to the root have equal cost. At that point, the level set in this command will determine which port the bridge will use. Set the priority value lower to increase the chance the interface will be used.

## Usage Examples

The following example sets the maximum priority on the Frame Relay sub-interface labeled 1.16:

(config)**#interface frame-relay 1.16**
(config-fr 1.16)**#spanning-tree priority 0**

# HDLC INTERFACE CONFIGURATION COMMAND SET

To activate the HDLC mode, enter the **interface hdlc** command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#interface hdlc 1
(config-hdlc 1)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:



All other commands for this command set are described in this section in alphabetical order.



# access-policy <*name*>

Use the **access-policy** command to assign a specified access policy to an interface. Access policies are applied to traffic entering an interface. Use the **no** form of this command to remove an access policy association. For more information on using access policies, refer to *ip policy-class <name>* on page 549.


*Configured access policies will only be active if the **ip firewall** command has been entered at the Global Configuration mode prompt to enable the AOS security features. All configuration parameters are valid, but no security data processing will be attempted unless the security features are enabled.*

## Syntax Description

| | |
|---|---|
| *<name>* | Identifies the configured access policy by alphanumeric descriptor (all access policy descriptors are case-sensitive). |

## Default Values

By default, there are no configured access policies associated with an interface.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include Ethernet sub-interfaces. |
| Release 6.1 | Command was expanded to include VLAN interfaces. |

## Functional Notes

To assign an access policy to an interface, enter the interface configuration mode for the desired interface and enter **access-policy** <*policy name*>.

## Usage Examples

The following example associates the access policy **Private** (to allow inbound traffic to the Web server) to the HDLC interface 1:

Enable the AOS security features:
(config)**#ip firewall**

Associate the access policy with the HDLC interface 1:

(config)**#interface hdlc 1**
(config-hdlc 1)**#access-policy Private**

# alias link<*"text"*>

Each configured HDLC interface (when referenced using SNMP) contains a link (physical port) and a bundle (group of links). RFC1471 (for Link Connection Protocol) provides an interface table to manage lists of bundles and associated links. The **alias link** command provides the management station an identifying description for each link (HDLC physical).

## Syntax Description

| | |
|---|---|
| <*"text"*> | Describes the interface (for SNMP) using an alphanumeric character string enclosed in quotation marks. |

## Default Values

By default, the HDLC identification string appears as empty quotes. (" ")

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Functional Notes

The **alias link** string should be used to uniquely identify an HDLC link. Enter a string that clearly identifies the link.

## Usage Examples

The following example defines a unique character string for the virtual HDLC interface (1):

(config)**#interface hdlc 1**
(config-hdlc 1)**#alias link "HDLC_link_1"**

## Technology Review

Please refer to RFC1990 for a more detailed discussion on HDLC links and bundles.

# bandwidth *<value>*

Use the **bandwidth** command to provide the bandwidth value of an interface to the higher-level protocols. This value is used in cost calculations. Use the **no** form of this command to restore the default values.

## Syntax Description

| | |
|---|---|
| *<value>* | Specifies the bandwidth in kbps. Range is 1 to 4,294,967,295 kbps. |

## Default Values

To view default values use the **show interfaces** command.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Functional Notes

The **bandwidth** command is an informational value that is communicated to the higher-level protocols to be used in cost calculations. This is a routing parameter only and does not affect the physical interface.

## Usage Examples

The following example sets bandwidth of the HDLC interface to 10 Mbps:

(config)**#interface hdlc 1**
(config-hdlc 1)**#bandwidth 10000**

# bridge-group *<value>*

Use the **bridge-group** command to assign an interface to the specified bridge group. Use the **no** form of this command to remove the interface from the bridge group.

## Syntax Description

| | |
|---|---|
| *<value>* | Specifies the bridge group (by number) to which to assign this interface. Range is 1 to 255. |

## Default Values

By default, there are no configured bridge groups.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Functional Notes

A bridged network can provide excellent traffic management to reduce collisions and limit the amount of bandwidth wasted with unnecessary transmissions when routing is not necessary. Any two interfaces can be bridged (Ethernet to T1 bridge, Ethernet to Frame Relay sub-interface, etc.).

## Usage Examples

The following example assigns the HDLC interface labeled 1 to bridge group 1:

(config)**#interface hdlc 1**
(config-hdlc 1)**#bridge-group 1**

# crypto map *<name>*

Use the **crypto map** command to associate crypto maps with the interface. Use the **no** form of this command to remove a crypto map from an interface.

> NOTE *When you apply a map to an interface, you are applying all crypto maps with the given map name. This allows you to apply multiple crypto maps if you have created maps which share the same name but have different map index numbers.*

> NOTE *For VPN configuration example scripts, refer to the technical support note **Configuring VPN** located on the ADTRAN OS System Documentation CD provided with your unit.*

## Syntax Description

| | |
|---|---|
| *<name>* | Enter the crypto map name that you wish to assign to the interface. |

## Default Values

By default, no crypto maps are assigned to an interface.

## Command History

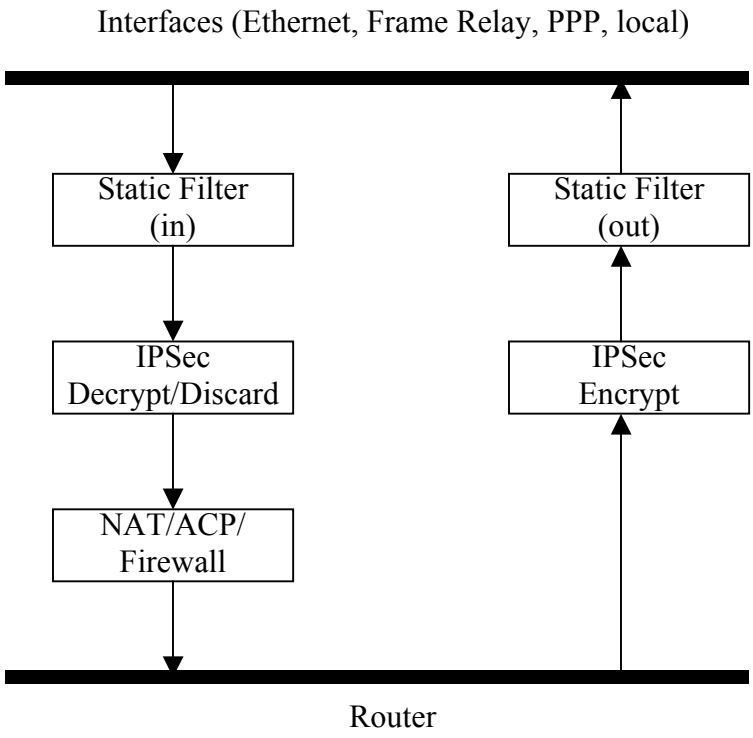
| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Functional Notes

When configuring a system to use both the stateful inspection firewall and IKE negotiation for VPN, keep the following notes in mind.

When defining the policy-class and associated access-control lists (ACLs) that describe the behavior of the firewall, do not forget to include the traffic coming into the system over a VPN tunnel terminated by the system. The firewall should be set up with respect to the un-encrypted traffic that is destined to be sent or received over the VPN tunnel. The following diagram represents typical AOS data-flow logic.




As shown in the diagram above, data coming into the product is first processed by the static filter associated with the interface on which the data is received. This access-group is a true static filter and is available for use regardless of whether the firewall is enabled or disabled. Next (if the data is encrypted) it is sent to the IPSec engine for decryption. The decrypted data is then processed by the stateful inspection firewall. Therefore, given a terminating VPN tunnel, only un-encrypted data is processed by the firewall.

The ACLs for a crypto map on an interface work in reverse logic to the ACLs for a policy-class on an interface. When specifying the ACLs for a crypto map, the source information is the private local-side, un-encrypted source of the data. The destination information will be the far-end, un-encrypted destination of the data. However, ACLs for a policy-class work in reverse. The source information for the ACL in a policy-class is the far-end. The destination information is the local-side.

## Usage Examples

The following example applies all crypto maps with the name **MyMap** to the HDLC 1 interface:

(config)**#interface hdlc 1**
(config-hdlc 1)**#crypto map MyMap**

# dial-backup auto-backup

Use the **dial-backup auto-backup** command to configure the interface to automatically attempt a dial-backup upon failure. Use the **no** form of this command to disable this feature. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of *dial-backup call-mode* on page 1259.

## Syntax Description

No subcommands.

## Default Values

By default, all backup endpoints will automatically attempt dial-backup upon a failure.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include the PPP interface. |

## Usage Examples

The following example enables automatic dial-backup on the endpoint:

(config)**#interface hdlc 1**
(config-hdlc 1)**#dial-backup auto-backup**

# dial-backup auto-restore

Use the **dial-backup auto-restore** command to configure the interface to automatically discontinue dial-backup when all network conditions are operational. Use the **no** form of this command to disable the auto-restore feature. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of *dial-backup call-mode* on page 1259.

## Syntax Description

No subcommands.

## Default Values

By default, all backup endpoints will automatically restore the primary connection when the failure condition clears.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include the PPP interface. |

## Usage Examples

The following example configures the AOS to automatically restore the primary connection when the failure condition clears:

(config)**#interface hdlc 1**
(config-hdlc 1)**#dial-backup auto-restore**

# dial-backup backup-delay *<value>*

Use the **dial-backup backup-delay** command to configure the amount of time the router will wait after the failure condition is recognized before attempting to backup the link. Use the **no** form of this command to return to the default value. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of *dial-backup call-mode* on page 1259.

## Syntax Description

| | |
|---|---|
| *<value>* | Specifies the delay period (in seconds) a failure must be active before the AOS will enter backup operation on the interface. Range is 10 to 86,400 seconds. |

## Default Values

By default, the **dial-backup backup-delay** period is set to 10 seconds.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include the PPP interface. |

## Usage Examples

The following example configures the AOS to wait **60** seconds (on an endpoint with an active alarm condition) before attempting dial-backup operation:

(config)**#interface hdlc 1**
(config-hdlc 1)**#dial-backup backup-delay 60**

# dial-backup call-mode

Use the **dial-backup call-mode** command to specify whether the configured backup interface answers or originates (or a combination of both ) backup calls. Use the **no** form of this command to return to the default value. Variations of this command include:

**dial-backup call-mode answer**
**dial-backup call-mode answer-always**
**dial-backup call-mode originate**
**dial-backup call-mode originate-answer**
**dial-backup call-mode originate-answer-always**

## Syntax Description

| | |
|---|---|
| **answer** | Answers and backs up primary link on failure. |
| **answer-always** | Answers and backs up regardless of primary link state. |
| **originate** | Originates backup call on primary link failure. |
| **originate-answer** | Originates or answers call on primary link failure. |
| **originate-answer-always** | Originates on failure; answers and backs up always. |

## Default Values

By default, the **dial-backup call-mode** is set to **originate-answer**.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include the PPP interface. |

## Functional Notes

The majority of the configuration for AOS dial-backup implementation is configured via the dial-backup PPP interface configuration commands. However, the numbers dialed are configured in the primary interface. Full sample configurations follow:

**Sample configuration for remote router (dialing out)**

```
hostname “Remote3200”
enable password adtran
!
interface eth 0/1
  ip address 192.168.1.254 255.255.255.0
  no shutdown
!
interface modem 1/3
 no shutdown
!
interface t1 1/1
```

```
  coding b8zs
  framing esf
  clock source line
  tdm-group 1 timeslots 1-24
  no shutdown
!
interface fr 1 point-to-point
  frame-relay lmi-type ansi
  no shutdown
  cross-connect 1 t1 1/1 1 fr 1
!
interface fr 1.16 point-to-point
  frame-relay interface-dlci 16
  ip address 10.1.1.2 255.255.255.252
  dial-backup call-mode originate
  dial-backup number 5551111 analog ppp1
  dial-backup number 5552222 analog ppp1
   no shutdown
!
interface ppp 1
  ip address  172.22.56.1  255.255.255.252
  ppp authentication chap
 username remoterouter password remotepass
 ppp chap hostname localrouter
  ppp chap password adtran
  no shutdown
!
ip route 192.168.2.0 255.255.255.0 172.22.56.2 255.255.255.252
!
line telnet 0 4
  password password
```

**Sample configuration for central router (dialing in)**

```
hostname “Central3200”
enable password adtran
!
interface eth 0/1
  ip address  192.168.100.254  255.255.255.0
  no shutdown
!
interface modem 1/3
 no shutdown
!
interface t1 1/1
```

```
  coding b8zs
  framing esf
  clock source line
  tdm-group 1 timeslots 1-24
  no shutdown
!
interface fr 1 point-to-point
  frame-relay lmi-type ansi
  no shutdown
  cross-connect 1 t1 1/1 1 fr 1
!
interface fr 1.100 point-to-point
  frame-relay interface-dlci 100
  ip address 10.1.1.1 255.255.255.252
  dial-backup call-mode answer
  dial-backup number 555-8888 analog ppp 1
!
interface ppp 1
  ip address  172.22.56.2 255.255.255.252
  ppp authentication chap
 username localrouter password adtran
 ppp chap hostname remoterouter
 ppp chap password remotepass
  no shutdown
!
ip route 192.168.1.0 255.255.255.0 172.22.56.1 255.255.255.252

line telnet 0 4
  password password
```

## Usage Examples

The following example configures AOS to generate backup calls for this endpoint using an analog modem interface (to phone number 555 1111) but never answer calls and specifies **ppp 1** as the backup interface:

```
(config)#interface hdlc 1
(config-hdlc 1)#backup call-mode originate
(config-hdlc 1)#backup number 555 1111 analog ppp 1
```

## Technology Review

This technology review provides information regarding specific dial-backup router behavior (i.e., when the router will perform dial-backup, where in the configuration the AOS accesses specific routing information, etc.):

### Dialing Out

1. The AOS determines to place an outbound call when either the Layer 1 or Layer 2 has a failure.
2. When placing outbound calls, the AOS matches the number dialed to a PPP interface. This is accomplished with an addition to the dial-backup number command (refer to *dial-backup number <number>* on page 1266).
3. When placing the call, the AOS uses the configuration of the related PPP interface for authentication and IP negotiation.
4. If the call fails to connect on the first number dialed, the AOS places a call to the second number (if a second number is configured). The second number to be dialed references a separate PPP interface.

### Dialing In

1. The AOS receives an inbound call on a physical interface.
2. Caller ID is used to match the **dial-backup number** command to the configured PPP interface.
3. If a match is found, the call connects and the AOS pulls down the primary connection if it is not already in a down state.
4. If no match is found from caller ID, the call is terminated.

# dial-backup connect-timeout *<value>*

Use the **dial-backup connect-timeout** command to specify the number of seconds to wait for a connection after a call is attempted before trying to call again or dialing a different number. It is recommended this number be greater than 60. Use the **no** form of this command to return to the default setting. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of the command *dial-backup call-mode* on page 1259.

## Syntax Description

| | |
|---|---|
| *<value>* | Selects the amount of time (in seconds) that the router will wait for a connection before attempting another call. Valid range is 10 to 300 seconds. |

## Default Values

By default, the **dial-backup connect-timeout** period is set to 60 seconds.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include the PPP interface. |

## Usage Examples

The following example configures the AOS to wait **120** seconds before retrying a failed dial-backup call:

```
(config)#interface hdlc 1
(config-hdlc 1)#dial-backup connect-timeout 120
```

# dial-backup force

Use the **dial-backup force** command to manually override the automatic dial-backup feature. This can be used to force a link into backup to allow maintenance to be performed on the primary link without disrupting data. Use the **no** form of this command to return to the normal dial-backup operation state. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of the command *dial-backup call-mode* on page 1259. Variations of this command include:

**dial-backup force backup**
**dial-backup force primary**

## Syntax Description

| | |
|---|---|
| **backup** | Forces backup regardless of primary link state. |
| **primary** | Forces primary link regardless of its state. |

## Default Values

By default, this feature is disabled.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include the PPP interface. |

## Usage Examples

The following example configures the AOS to force this interface into dial-backup:

```
(config)#interface hdlc 1
(config-hdlc 1)#dial-backup force backup
```

# dial-backup maximum-retry <*value*>

Use the **dial-backup maximum-retry** command to select the number of calls the router will make when attempting to backup a link. Use the **no** form of this command to return to the default state. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of the command *dial-backup call-mode* on page 1259.

## Syntax Description

| | |
|---|---|
| <*value*> | Selects the number of call retry attempts that will be made after a link failure. Valid range is 0 to 15 attempts.<br>Setting this value to 0 will allow unlimited retries during the time the network is failed. |

## Default Values

By default, **dial-backup maximum-retry** is set to 0 attempts.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include the PPP interface. |

## Usage Examples

The following example configures the AOS to retry a dial-backup call four times before considering backup operation not available:

(config)**#interface hdlc 1**
(config-hdlc 1)**#dial-backup maximum-retry 4**

# dial-backup number <*number*>

Use the **dial-backup number** command to configure the phone number and the call type the router will dial upon network failure. Multiple entries can be made for an interface to allow alternate sites to be dialed. Use the **no** form of this command to remove a configured dial-backup number. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of the command *dial-backup call-mode* on page 1259. Variations of this command include:

**dial-backup number** <*number*> **analog ppp** <*interface*>
**dial-backup number** <*number*> **digital-56k** <*isdn min chan*> <*isdn max chan*> **ppp** <*interface*>
**dial-backup number** <*number*> **digital 64k** <*isdn min chan*> <*isdn max chan*> **ppp** <*interface*>

## Syntax Description

| | |
|---|---|
| <*number*> | Specifies the phone numbers to call when the backup is initiated. |
| **analog ppp** | Indicates the number connects to an analog modem. |
| **digital-56k** | Indicates the number belongs to a digital 56 kbps per DS0 connection. |
| **digital-64k** | Indicates the number belongs to a digital 64 kbps per DS0 connection. |
| <*isdn min chan*> | Specifies the minimum number of DS0s required for a digital 56 or 64 kbps connection. Range is 1 to 24 DS0s. |
| <*isdn max chan*> | Specifies the maximum number of DS0s desired for a digital 56 or 64 kbps connection. Range is 1 to 24 DS0s. |
| **ppp** <*interface*> | Specifies the PPP interface to use as the backup for this interface. For example, **ppp 1**. |

## Default Values

By default, there are no configured dial-backup numbers.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include the PPP interface. |

## Usage Examples

The following example configures the AOS to dial 704-555-1212 (digital 64 kbps connection) to initiate dial-backup operation for this endpoint using the configured **ppp 1** backup interface:

(config)**#interface hdlc 1**
(config-hdlc 1)**#dial-backup number 7045551212 digital-64k 1 1 ppp 1**

# dial-backup priority <*value*>

Use the **dial-backup priority** command to select the backup priority for this interface. This allows the user to establish the highest priority backup link and ensure that link will override backups attempted by lower priority links. Use the **no** form of this command to return to the default value. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of the command *dial-backup call-mode* on page 1259.

## Syntax Description

| | |
|---|---|
| <*value*> | Sets the relative priority of this link. Valid range is 0 to 100. A value of 100 designates the highest priority. |

## Default Values

By default, **dial-backup priority** is set to 50.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include the PPP interface. |

## Usage Examples

The following example assigns the highest priority to this endpoint:

(config)**#interface hdlc 1**
(config-hdlc 1)**#dial-backup priority 100**

# dial-backup randomize-timers

Use the **dial-backup randomize-timers** command to randomize the call timers to minimize potential contention for resources. Use the **no** form of this command to return to the default value. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of the command *dial-backup call-mode* on page 1259.

## Syntax Description

No subcommands.

## Default Values

By default, the AOS does not randomize the dial-backup call timers.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include the PPP interface. |

## Usage Examples

The following example configures the AOS to randomize the dial-backup timers associated with this endpoint:

(config)**#interface hdlc 1**
(config-hdlc 1)**#dial-backup randomize-timers**

# dial-backup redial-delay <*value*>

Use the **dial-backup redial-delay** command to configure the delay after an unsuccessful call until the call will be re-tried. Use the **no** form of this command to return to the default setting. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of the command *dial-backup call-mode* on page 1259.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the delay in seconds between attempting to re-dial a failed backup attempt. Range is 10 to 3600 seconds. |

## Default Values

By default, **dial-backup redial-delay** is set to 10 seconds.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include the PPP interface. |

## Usage Examples

The following example configures a redial delay of **25** seconds on this endpoint:

(config)**#interface hdlc 1**
(config-hdlc 1)**#dial-backup redial-delay 25**

# dial-backup restore-delay *<value>*

Use the **dial-backup restore-delay** command to configure the amount of time the router will wait after the network is restored before disconnecting the backup link and reverting to the primary. This setting is used to prevent disconnecting the backup link if the primary link is "bouncing" in and out of alarm. Use the **no** form of this command to return to the default setting. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of the command *dial-backup call-mode* on page 1259.

## Syntax Description

| | |
|---|---|
| *<value>* | Specifies the number of seconds the AOS will wait (after a primary link is restored) before disconnecting dial-backup operation. Range is 10 to 86,400 seconds. |

## Default Values

By default, **dial-backup restore-delay** is set to 10 seconds.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include the PPP interface. |

## Usage Examples

The following example configures the AOS to wait **30** seconds before disconnecting dial-backup operation and restoring the primary connection for this endpoint:

(config)**#interface hdlc 1**
(config-hdlc 1)**#dial-backup restore-delay 30**

# dial-backup schedule

Use the **dial-backup schedule** command to set the time of day that backup will be enabled. Use this command if backup is desired only during normal business hours and on specific days of the week. Use the **no** form of this command to disable dial-backup (as specified). For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of the command *dial-backup call-mode* on page 1259. Variations of this command include:

**dial-backup schedule day** <*name*>
**dial-backup schedule enable-time** <*value*>
**dial-backup schedule disable-time** <*value*>

## Syntax Description

| | |
|---|---|
| **day** <*name*> | Sets the days to allow backup. Valid range is Monday through Sunday. |
| **enable-time** <*value*> | Sets the time of day to enable backup. Time is entered in 24-hour format (00:00). |
| **disable-time** <*value*> | Sets the time of day to disable backup. Time is entered in 24-hour format (00:00). |

## Default Values

By default, dial-backup is enabled for all days and times if the dial-backup auto-backup command has been issued and the dial-backup schedule has not been entered.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include the PPP interface. |

## Usage Examples

The following example enables dial-backup Monday through Friday 8:00 am to 7:00 pm:

(config)**#interface hdlc 1**
(config-hdlc 1)**#dial-backup schedule enable-time 08:00**
(config-hdlc 1)**#dial-backup schedule disable-time 19:00**
(config-hdlc 1)**#no dial-backup schedule day Saturday**
(config-hdlc 1)**#no dial-backup schedule day Sunday**

# dial-backup shutdown

Use the **dial-backup shutdown** command to deactivate all dial-backup functionality in the unit. Dial-backup configuration parameters are kept intact, but the unit will not initiate (or respond) to dial-backup sequences in the event of a network outage. Use the **no** form of this command to reactivate the dial-backup interface. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of the command *dial-backup call-mode* on page 1259.

## Syntax Description

No subcommands.

## Default Values

By default, all AOS interfaces are disabled.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include the PPP interface. |

## Usage Examples

The following example deactivates the configured dial-backup interface:

(config)**#interface hdlc 1**
(config-hdlc 1)**#dial-backup shutdown**

# dynamic-dns

Use the **dynamic-dns** command to configure Dynamic DNS service provided by Dynamic Network Services, Inc. (www.dyndns.org). Use the **no** versions of these commands to disable these features. Variations of this command include:

**dynamic-dns custom** *<hostname> <value>*
**dynamic-dns dyndns** *<hostname> <username> <password>*
**dynamic-dns dyndns-custom** *<hostname> <username> <password>*
**dynamic-dns dyndns-static** *<hostname> <username> <password>*

## Syntax Description

| | |
|---|---|
| *<hostname>* | Specifies the host name for the server that updates the Dynamic Domain Name Server (DNS). |
| *<value>* | Specifies the intervals in minutes to update the server with information (updates also occur when the interface's IP address changes regardless of the update intervals).<br>Refer to *Functional Notes* below for additional argument descriptions. |

## Default Values

No default is necessary for this command.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |
| Release 12.1 | Command was expanded. |

## Functional Notes

**custom** - Constanttime.com's Custom Dynamic DNS$^{SM}$ service allows you complete access and management control over your domain name regardless of where you purchased/registered it. This allows you to manage IP address mappings (A records), domain aliases (CNAME records) and mail servers (MX records).

**dyndns** - The Dynamic DNS$^{SM}$ offered by Dynamic Network Services Inc., (DynDNS.org) allows you to alias a dynamic IP address to a static host name in various domains. This allows your unit to be more easily accessed from various locations on the Internet. This service is provided for up to five host names.

**dyndns-custom -** DynDNS.org's Custom DNS$^{SM}$ service provides a full DNS solution, giving you complete control over an entire domain name. A web-based interface provides two levels of control over your domain, catering to average or advanced users. Five globally redundant DNS servers ensure that your domain will always resolve.

A choice of two interfaces is available. The basic interface is designed for most users. It comes preconfigured for the most common configuration and allows for easy creation of most common record types. The advanced interface is designed for system administrators with a solid DNS background, and provides layout and functionality similar to a BIND zone file allowing for the creation of nearly any record type.

Custom DNS$^{SM}$ can be used with both static and dynamic IPs and has the same automatic update capability through Custom DNS-aware clients as Dynamic DNS.

**dyndns-static -** The Static DNS service is similar to DynDNS.org's Dynamic DNS$^{SM}$ service in that it allows a host name such as yourname.dyndns.org to point to your IP address. Unlike a Dynamic DNS host, a Static DNS host does not expire after 35 days without updates, but updates take longer to propagate though the DNS system. This service is provided for up to five host names.

If your IP address does not change often or at all but you still want an easy name to remember it by (without having to purchase your own domain name), Static DNS service is ideal for you.

If you would like to use your own domain name (such as yourname.com), you need Custom DNS service which also provides full dynamic and static IP address support.

## Usage Examples

The following example sets the dynamic-dns to **dyndns-custom** with host name **host**, user name **user**, and password **pass**:

(config)**#interface hdlc 1**
(config-hdlc 1)**#dynamic-dns dyndns-custom host user pass**

# fair-queue

Use the **fair-queue** command to enable weighted fair queuing (WFQ) on an interface. Use the **no** form of this command to disable WFQ and enable FIFO (first-in-first-out) queueing for an interface. WFQ is enabled by default for WAN interfaces. Variations of this command:

**fair-queue**
**fair-queue** <*value*>

## Syntax Description

| | |
|---|---|
| <*value*> | Optional. Value that specifies the maximum number of packets that can be present in each conversation sub-queue. Packets received for a conversation after this limit is reached are discarded. Range is 16 to 512 packets. |

## Default Values

By default, fair-queue is enabled with a threshold of 64 packets.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Usage Examples

The following example enables WFQ on the interface with a threshold set at 100 packets:

(config)**#interface hdlc 1**
(config-hdlc 1)**#fair-queue 100**

# hold-queue <*value*> out

Use the **hold-queue** command to change the overall size of an interface's WAN output queue. Use the **no** form of this command to return to the default settings.

## Syntax Description

| | |
|---|---|
| <*value*> | The total number of packets the output queue can contain before packets are dropped. Range is 16 to 1000 packets. |

## Default Values

The default queue size for WFQ is 400. The default queue size for PPP FIFO and Frame Relay round-robin is 200.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Usage Examples

The following example sets the overall output queue size to **700**:

(config)**#interface hdlc 1**
(config-hdlc 1)**#hold-queue 700**

# ip access-group *<name>*

Use the **ip access-group** command to create an access control list to be used for packets transmitted on or received from the specified interface. Use the **no** form of this command to disable this type of control. Variations of this command include:

**ip access-group** <*name*> **in**
**ip access-group** <*name*> **out**

## Syntax Description

| | |
|---|---|
| *<name>* | Assigned IP access control list name. |
| **in** | Enables access control on packets received on the specified interface. |
| **out** | Enables access control on packets transmitted on the specified interface. |

## Default Values

By default, these commands are disabled.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Functional Notes

When this command is enabled, the IP destination address of each packet must be validated before being passed through. If the packet is not acceptable per these settings, it is dropped.

## Usage Examples

The following example sets up the unit to only allow Telnet traffic (as defined in the user-configured **TelnetOnly** IP access control list) into the HDLC interface:

(config)**#ip access-list extended TelnetOnly**
(config-ext-nacl)**#permit tcp any any eq telnet**
(config-ext-nacl)**#int hdlc 1**
(config-hdlc 1)**#ip access-group TelnetOnly in**

# ip address *<ip address> <subnet mask>*

Use the **ip address** command to define an IP address on the specified interface. Use the **no** form of this command to remove a configured IP address. Variations of this command include:

**ip address** *<ip address> <subnet mask>*
**ip address** *<ip address> <subnet mask>* **secondary**

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies a valid IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| *<subnet mask>* | Specifies the subnet mask that corresponds to a range of IP addresses (network) or a specific host. Subnet masks can be expressed in dotted decimal notation (for example, **255.255.255.0**) or as a prefix length (for example, **/24**). |
| **secondary** | Optional. Configures a secondary IP address for the specified interface. |

## Default Values

By default, there are no assigned IP addresses.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced |

## Functional Notes

Use secondary IP addresses to allow dual subnets on a single interface (when you need more IP addresses than the primary subnet can provide). When using secondary IP addresses, avoid routing loops by verifying that all devices on the network segment are configured with secondary IP addresses on the secondary subnet.

## Usage Examples

The following example configures a secondary IP address of **192.22.72.101/30**:

(config)**#hdlc 1**
(config-hdlc 1)**#ip address 192.22.72.101 255.255.255.252 secondary**

# ip ffe

Use the **ip ffe** command to enable the FastFlow Engine (FFE) on this interface with the default number of entries. Use the **no** form of these commands to disable the FFE. Variations of this command include:

**ip ffe max-entries** <*entries*>

*Issuing this command will cause all FFE entries on this interface to be cleared.*

## Syntax Description

| | |
|---|---|
| **max-entries** <*entries*> | Specifies the maximum number of entries stored in the flow table. Valid range is from 1 to 8192. |

## Default Values

By default, the FFE is disabled. The default number of **max-entries** is 4096.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Functional Notes

The FFE can be used to help reduce routing overhead and thus reduce overall routing times. Routing times are reduced by the creation of a flow table on the ingress interface. The maximum number of entries that can be stored in the flow table at any one time may be specified by using the **max-entries** subcommand. When features such as firewall, VPN, and static filters are enabled, routing time reduction may not be realized.

## Usage Examples

The following example enables the FFE and sets the maximum number of entries in the flow table to 50:

(config)**#hdlc 1**
(config-hdlc 1)**#ip ffe max-entries 50**

## Technology Review

The FastFlow system goal is to increase IP packet throughput by moving as much of the packet processing into the engine as possible. Packets are classified into flows based upon the IP protocol (TCP, UDP, ICMP, etc.), the source and destination IP addresses, IP TOS, and the protocol-specific information such as the source and destination port numbers. Flows are defined as the unidirectional representation of a conversation between two IP hosts. Each ingress interface keeps its own flow table, a collection of flow entries.

The first packet in a flow that is forwarded through the unit will build a flow entry. When a flow entry is looked up but no entry is found, a FastFlowBuilder object is allocated and attached to the packet. As the packet passes through the various processing layers, each subsystem will add processing to the FastFlowBuilder. When packet is about to be forwarded out of the egress interface, the FastFlowBuilder will be finalized. That is, the flow entry being built will be checked for completeness and committed to the flow table on the ingress interface. Subsequent flow matches can then bypass the normal processing layers.

# ip helper-address *<ip address>*

Use the **ip helper-address** command to configure the AOS to forward User Datagram Protocol (UDP) broadcast packets received on the interface. Use the **no** form of this command to disable forwarding packets.

> NOTE *The **ip helper** command must be used in conjunction with the **ip forward-protocol** command to configure the AOS to forward UDP broadcast packets. Refer to ip forward-protocol udp <value> on page 532 for more information.*

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies the destination IP address for the forwarded UDP packets. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

## Default Values

By default, broadcast UDP packets are not forwarded.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced |

## Functional Notes

When used in conjunction with the **ip forward-protocol** command, the **ip helper-address** feature allows you to customize which broadcast packets are forwarded.

To implement the helper address feature, assign a helper-address(es) (specifying the device that needs to receive the broadcast traffic) to the interface closest to the host that transmits the broadcast packets. When broadcast packets (of the specified type forwarded using the **ip forward-protocol** command) are received on the interface, they will be forwarded to the device that needs the information.

Only packets meeting the following criteria are considered eligible by the **ip helper-address** feature:

1. The packet IP protocol is User Datagram Protocol (UDP).
2. Any UDP port specified using the **ip forward-protocol** command.
3. The media access control (MAC) address of the frame is an all-ones broadcast address (ffff.ffff.ffff).
4. The destination IP address is broadcast defined by all ones (255.255.255.255) or a subnet broadcast (for example, 192.33.4.251 for the 192.33.4.248/30 subnet).

## Usage Examples

The following example forwards all DNS broadcast traffic to the DNS server with IP address **192.33.5.99**:

(config)**#ip forward-protocol udp domain**
(config)**#interface hdlc 1**
(config-hdlc 1)**#ip helper-address 192.33.5.99**

# ip igmp

Use the **ip igmp** command to configure multicasting-related functions for the interface. Variations of this command include:

**ip igmp immediate-leave**
**ip igmp last-member-query-interval** <*milliseconds*>
**ip igmp querier-timeout** <*seconds*>
**ip igmp query-interval** <*seconds*>
**ip igmp query-max-response-time** <*seconds*>
**ip igmp static-group** <*address*>
**ip igmp version [1 | 2]**

## Syntax Description

| | |
|---|---|
| **immediate-leave** | Specifies that if only one host (or IGMP snooping switch) is connected to the interface, when a leave is received, multicast of that group is immediately terminated as opposed to sending a group query and timing out the group if no device responds. Works in conjunction with **ip igmp last-member-query-interval**. Applies to all groups when configured. Use the **no** form of this command to disable the **immediate-leave** feature. |
| **last-member-query-interval** <*milliseconds*> | Controls the timeout (in milliseconds) used to detect whether any group receivers remain on an interface after a receiver leaves a group. If a receiver sends a leave-group message (IGMP Version 2), the router sends a group-specific query on that interface. After twice the time specified by this command plus as much as one second longer, if no receiver responds, the router removes that interface from the group and stops sending that group's multicast packets to the interface. Range is 100 to 65,535 ms. Use the **no** form of this command to return to the default setting. |
| **querier-timeout** <*seconds*> | Specifies the interval (in seconds) that the router waits after the current querier's last query before it takes over as querier (IGMP V2). Range is 60 to 300 seconds. Use the **no** form of this command to return to the default setting. |
| **query-interval** <*seconds*> | Specifies the interval (in seconds) at which IGMP queries are sent on an interface. Host query messages are addressed to the all-hosts multicast group with an IP TTL of 1. The router uses queries to detect whether multicast group members are on the interface and to select an IGMP designated router for the attached segment (if more than one multicast router exists). Only the designated router for the segment sends queries. For IGMP V2, the designated router is the router with the lowest IP address on the segment. Range is 0 to 65,535 seconds. Use the **no** form of this command to return to the default setting. |
| **query-max-response-time** <*seconds*> | Specifies the maximum response time (in seconds) advertised by this interface in queries when using IGMP V2. Hosts are allowed a random time within this period to respond, reducing response bursts. Use the **no** form of this command to return to the default setting. |

## Syntax Description

| | |
|---|---|
| **static-group** <*address*> | Configures the router's interface to be a statically-connected member of the specified group. Packets received on the correct RPF interface are forwarded to this interface regardless of whether any receivers have joined the specified group using IGMP. Use the **no** form of this command to remove a configured static group. |
| **version [1 \| 2]** | Sets the interface's IGMP version. Use the **no** form of this command to return to the default setting. |

## Default Values

| | |
|---|---|
| **ip igmp immediate-leave** | No default |
| **ip igmp last-member-query-interval** | 1000 milliseconds |
| **ip igmp querier-timeout** | 2x the **query-interval** value |
| **ip igmp query-interval** | 60 seconds |
| **ip igmp query-max-response-time** | 10 seconds |
| **ip igmp static-group** | No default |
| **ip igmp version** | Version 1 |

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Usage Examples

The following example sets the query message interval on the interface to **200** milliseconds:

(config)**#interface hdlc 1**
(config-hdlc 1)**#ip igmp last-member-query-interval 200**

# ip mcast-stub downstream

Use the **ip mcast-stub downstream** command to enable multicast forwarding and IGMP (router mode) on an interface and place it in multicast stub downstream mode. Use the **no** form of this command to disable.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Functional Notes

This command is used in IP multicast stub applications in conjunction with the **ip mcast-stub helper-address** and **ip mcast-stub upstream** commands. Downstream interfaces connect to segments with multicast hosts. Multiple interfaces may be configured in downstream mode; however, interfaces connecting to the multicast network (upstream) should not be configured in downstream mode. Interfaces configured as downstream should have the lowest IP address of all IGMP-capable routers on the connected segment in order to be selected as the designated router and ensure proper forwarding. Refer to *ip mcast-stub helper-address <ip address>* on page 546 and *ip mcast-stub upstream* on page 1287 for more information.

## Usage Examples

The following example enables multicast forwarding and IGMP on the HDLC interface:

```
(config)#interface hdlc 1
(config-hdlc 1)#ip mcast-stub downstream
```

# ip mcast-stub fixed

Use the **ip mcast-stub fixed** command to allow forwarding of multicast traffic on a selected interface after enabling multicast routing. Use the **no** form of this command to disable this mode.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Functional Notes

Multicast routing must be enabled prior to setting **ip mcast-stub fixed** on the selected interface. Also, use the **ip igmp static-group** <*A.B.C.D*> command to receive multicast traffic without host-initiated Internet Group Management Protocol (IGMP) activity on the selected interface. Otherwise, all host-initiated IGMP transactions will enter multicast routes on the router's interface involved with IGMP activities.

### Usage Examples

The following example enables multicast traffic forwarding and IGMP on the interface:

(config)**#interface hdlc 1**
(config-hdlc 1)**#ip mcast-stub fixed**

# ip mcast-stub helper-enable

Use the **ip mcast-stub helper-enable** command to assign the **ip mcast-stub helper-address** as the IGMP proxy. Use the **no** form of this command to disable.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |
| Release 10.1 | Command was expanded to include HDLC interfaces. |

## Functional Notes

This command is used in IP multicast stub applications in conjunction with the **ip mcast-stub helper-address**, **ip mcast-stub upstream**, and **ip mcast-stub downstream** commands. When enabled, the interface becomes a helper forwarding interface. The IGMP host function is dynamically enabled and the interface becomes the active upstream interface, enabling the unit to perform as an IGMP proxy. Refer to *ip mcast-stub helper-address <ip address>* on page 546, *ip mcast-stub downstream* on page 1284, and *ip mcast-stub upstream* on page 1287 for more information.

## Usage Examples

The following example sets the helper address as the IGMP proxy:

(config)**#interface hdlc 1**
(config-hdlc 1)**#ip mcast-stub helper-enable**

# ip mcast-stub upstream

Use the **ip mcast-stub upstream** command to enable multicast forwarding on an interface and place it in multicast stub upstream mode. Use the **no** form of this command to disable.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Functional Notes

This command is used in IP multicast stub applications in conjunction with the **ip mcast-stub helper-address** and **ip mcast-stub downstream** commands. When enabled, the interface becomes a candidate to be a helper forwarding interface. If chosen as the best path toward the helper address by the router's unicast route table, the IGMP host function is dynamically enabled and the interface becomes the active upstream interface, enabling the router to perform as an IGMP proxy. Though multiple interfaces may be candidates, no more than one interface will actively serve as the helper forwarding interface. Refer to *ip mcast-stub helper-address <ip address>* on page 546 and *ip mcast-stub downstream* on page 1284 for more information.

## Usage Examples

The following example enables multicast forwarding on the HDLC interface:

(config)**#interface hdlc 1**
(config-hdlc 1)**#ip mcast-stub upstream**

# ip ospf

Use the **ip ospf** command to customize OSPF settings (if needed). Use the **no** form of these commands to return to the default settings. Variations of this command include:

**ip ospf authentication-key** <*password*>
**ip ospf cost** <*value*>
**ip ospf dead-interval** <*seconds*>
**ip ospf hello-interval** <*seconds*>
**ip ospf message-digest-key [1 | 2] md5** <*key*>
**ip ospf priority** <*value*>
**ip ospf retransmit-interval** <*seconds*>
**ip ospf transmit-delay** <*seconds*>

## Syntax Description

| | |
|---|---|
| **authentication-key** <*password*> | Assigns a simple-text authentication password to be used by other routers using the OSPF simple password authentication. |
| **cost** <*value*> | Specifies the OSPF cost of sending a packet on the interface. This value overrides any computed cost value. Range is 1 to 65,535. |
| **dead-interval** <*seconds*> | Sets the maximum interval (in seconds) allowed between hello packets. If the maximum is exceeded, neighboring devices will determine that the device is down. Range is 0 to 32,767 seconds. |
| **hello-interval** <*seconds*> | Specifies the interval (in seconds) between hello packets sent on the interface. Range is 0 to 32,767 seconds. |
| **message-digest-key [1 \| 2] md5** <*key*> | Configures OSPF Message Digest 5 (MD5) authentication (16-byte max) keys. |
| **priority** <*value*> | Sets the OSPF priority. The value set in this field helps determine the designated router for this network. Range is 0 to 255. |
| **retransmit-interval** <*seconds*> | Specifies the interval (in seconds) between link-state advertisements (LSAs). Range is 0 to 32,767 seconds. |
| **transmit-delay** <*seconds*> | Sets the estimated time (in seconds) required to send an LSA on the interface. Range is 0 to 32,767 seconds. |

## Default Values

| | |
|---|---|
| **dead-interval** <*seconds*> | 40 seconds |
| **hello-interval** <*seconds*> | 10 seconds: Ethernet, point-to-point, Frame Relay, Tunnel, and PPP |
| **retransmit-interval** <*seconds*> | 5 seconds |
| **transmit-delay** <*seconds*> | 1 second |

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Usage Examples

The following example sets the maximum number of seconds allowed between hello packets to 25,000:

(config)**#interface hdlc 1**
(config-hdlc 1)**#ip ospf dead-interval 25000**

# ip ospf authentication

Use the **ip ospf authentication** command to authenticate an interface that is performing OSPF authentication. Use the **no** form of this command to return to the default setting. Variations of this command include:

**ip ospf authentication**
**ip ospf authentication message-digest**
**ip ospf authentication null**

## Syntax Description

| | |
|---|---|
| **message-digest** | Optional. Select message-digest authentication type. |
| **null** | Optional. Specifies that no authentication be used. |

## Default Values

By default, this is set to null (meaning no authentication is used).

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Usage Examples

The following example specifies that no authentication will be used on the HDLC interface:

(config)**#interface hdlc 1**
(config-hdlc 1)**#ip ospf authentication null**

# ip ospf network

Use the **ip ospf network** command to specify the type of network on this interface. Use the **no** form of this command to return to the default setting. Variations of this command include:

**ip ospf network broadcast**
**ip ospf network point-to-point**

## Syntax Description

| | |
|---|---|
| **broadcast** | Set the network type for broadcast. |
| **point-to-point** | Set the network type for point-to-point. |

## Default Values

By default, Ethernet defaults to broadcast. All other interfaces default to point-to-point.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Functional Notes

A point-to-point network will not elect designated routers.

## Usage Examples

The following example designates a broadcast network type:

(config)**#interface hdlc 1**
(config-hdlc 1)**#ip ospf network broadcast**

# ip pim sparse-mode

Use the **ip pim sparse-mode** command to enable protocol-independent multicast (PIM) sparse mode for this interface. Use the **no** form of this command to disable PIM sparse mode.

## Syntax Description

No subcommands.

## Default Values

By default, PIM sparse mode for this interface is disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Functional Notes

PIM Sparse Mode is a multicast routing protocol that makes use of the unicast forwarding table. It builds unidirectional shared trees rooted at a Rendezvous Point (RP) for a multicast group or a shortest-path tree rooted at a specific source for a multicast group.

## Usage Examples

The following example enables PIM sparse mode on the HDLC 1 interface:

(config)**#interface hdlc 1**
(config-hdlc 1)**#ip pim sparse-mode**

# ip pim-sparse dr-priority <*value*>

Use the **ip pim-sparse dr-priority** command to specify the priority for the designated router (DR). This command modifies the routers priority in the DR election process. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the priority of this interface (to be used when determining the DR). Valid range is 1 to 4,294,967,295. |

## Default Values

By default, the priority of all PIM interfaces is 1.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Functional Notes

Interfaces advertise their configured priority values in the hello messages transmitted on the interface. Routers use the priority values to determine the appropriate DR. The router on the network segment with the highest priority is selected as the DR. If a hello message is received on the interface from a router on the network segment and it does not contain a priority, the entire network segment defaults to DR selection based on IP addresses instead of priority. In this instance, the DR is selected as the router on the network segment that has the highest IP address. AOS will always include a priority in all transmitted hello messages. If no priority is specifically designated by the user, the priority is set as the default of 1.

## Usage Examples

The following example specifies a priority of **100** on the HDLC 1 interface:

(config)**#interface hdlc 1**
(config-hdlc 1)**#ip pim-sparse dr-priority 5**

# ip pim-sparse hello-timer <*value*>

Use the **ip pim-sparse hello-timer** command to specify protocol-independent multicast (PIM) sparse hello timer period. This is the time interval at which periodic hellos are sent out on all interfaces of a PIM-capable router. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the interval (in seconds) at which periodic hellos are sent out of the interface. Valid range is 10 to 3600 seconds. |

## Default Values

By default, the hellos are transmitted on PIM interfaces every 60 seconds.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Functional Notes

Hello messages are used to inform neighbors of a router's presence. Hello messages normally generate a small amount of traffic on an interface. Setting the **hello-timer** to a small interval increases the number of hellos sent (thus increasing the amount of traffic). Set the **hello-timer** to a reasonable value, taking into consideration the bandwidth available on the interface.

## Usage Examples

The following example specifies hellos be sent on the HDLC 1 interface every **3600** seconds:

(config)**#interface hdlc 1**
(config-hdlc 1)**#ip pim-sparse hello-timer 3600**

# ip pim-sparse nbr-timeout <*value*>

Use the **ip pim-sparse nbr-timeout** command to specify protocol-independent multicast (PIM) sparse neighbor timeout. This is the time interval after which a PIM-capable router will consider a neighbor not present. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the time interval in seconds after which a neighbor is considered not present. Valid range is 30 to 10,800 seconds. |

## Default Values

By default, the nbr-timeout is set to 105 seconds.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example sets the nbr-timeout to **300** seconds:

(config)**#interface hdlc 1**
(config-hdlc 1)**#ip pim-sparse nbr-timeout 300**

# ip pim-sparse override-interval <*value*>

Use the **ip pim-sparse override-interval** command to specify the protocol-independent multicast (PIM) sparse join/prune override interval. This delay interval is the period after a join/prune that another router on the LAN may override a join/prune. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the delay time in milliseconds. Valid range is 0 to 65,535 milliseconds. |

## Default Values

By default, the override interval is set to 2500 milliseconds.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example sets the override interval to **3000** milliseconds:

(config)**#interface hdlc 1**
(config-hdlc 1)**#ip pim-sparse override-interval 3000**

# ip pim-sparse propagation-delay <*value*>

Use the **ip pim-sparse propagation-delay** command to specify the expected propagation delay for join/prune messages. Set the propagation delay (in milliseconds) to estimate the amount of delay found in the local link. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the expected propagation delay in the local link in milliseconds. Valid range is 0 to 32,767 milliseconds. |

## Default Values

By default, the propagation delay is set to 500 milliseconds.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example sets the expected propagation delay to **300** milliseconds on the HDLC 1 interface:

(config)**#interface hdlc 1**
(config-hdlc 1)**#ip pim-sparse propagation-delay 300**

# ip policy route-map *<name>*

Use the **ip policy route-map** command to assign a policy route-map to this interface. Use the **no** form of this command to remove the route-map policy.

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies the name of the policy route map to assign to this interface. |

## Default Values

By default, no policy route map is assigned to this interface.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example assigns the policy route map **policy1** to the HDLC 1 interface:

(config)**#interface hdlc 1**
(config-hdlc 1)**#ip policy route-map policy1**

# ip proxy-arp

Use the **ip proxy-arp** command to enable proxy Address Resolution Protocol (ARP) on the interface. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, proxy ARP is enabled.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Functional Notes

In general, the principle of proxy ARP allows a router to insert its IP address in the source IP address field of a packet (if the packet is from a host on one of its subnetworks). This allows hosts to reach devices on other subnetworks without implementing routing or specifying a default gateway.

If proxy ARP is enabled, the AOS will respond to all proxy ARP requests with its specified MAC address and forward packets accordingly.

Enabling proxy ARP on an interface may introduce unnecessary ARP traffic on the network.

## Usage Examples

The following enables proxy ARP on the HDLC interface:

(config)**#interface hdlc 1**
(config-hdlc 1)**#ip proxy-arp**

# ip rip receive version

Use the **ip rip receive version** command to configure the RIP version the unit accepts in all RIP packets received on the interface. Use the **no** form of this command to restore the default value. Variations of this command include:

**ip rip receive version 1**
**ip rip receive version 2**

## Syntax Description

| | |
|---|---|
| **1** | Only accept received RIP version 1 packets on the interface. |
| **2** | Only accept received RIP version 2 packets on the interface. |

## Default Values

By default, all interfaces implement RIP version 1 (the default value for the version command).

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Functional Notes

Use the **ip rip receive version** command to specify a RIP version that overrides the **version** (in the Router RIP) configuration. Refer to *version* on page 1734 for more information.

The AOS only accepts one version (either 1 or 2) on a given interface.

## Usage Examples

The following example configures the HDLC interface to accept only RIP version 2 packets:

(config)**#interface hdlc 1**
(config-hdlc 1)**#ip rip receive version 2**

# ip rip send version

Use the **ip rip send version** command to configure the RIP version the unit sends in all RIP packets transmitted on the interface. Use the **no** form of this command to restore the default value. Variations of this command include:

**ip rip send version 1**
**ip rip send version 2**

## Syntax Description

| | |
|---|---|
| **1** | Only transmits RIP version 1 packets on the interface. |
| **2** | Only transmits RIP version 2 packets on the interface. |

## Default Values

By default, all interfaces transmit RIP version 1 (the default value for the **version** command).

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Functional Notes

Use the **ip rip send version** command to specify a RIP version that overrides the **version** (in the Router RIP) configuration. Refer to *version* on page 1734 for more information.

The AOS only transmits one version (either 1 or 2) on a given interface.

## Usage Examples

The following example configures the HDLC interface to transmit only RIP version 2 packets:

(config)**#interface hdlc 1**
(config-hdlc 1)**#ip rip send version 2**

# ip rip summary-address *<ip address> <subnet mask>*

Use the **ip rip summary-address** command to manually summarize the routes Routing Information Protocol (RIP) will advertise and send out a specified interface. Use the **no** form of this command to disable this mode.

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies the summarized network IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.0**). |
| *<subnet mask>* | Specifies the subnet mask that corresponds to the range of IP addresses (network). Subnet masks can be expressed in dotted decimal notation (for example, **255.255.255.0**) or as a prefix length (for example, **/24**). |

## Default Values

By default, no manual summarization is applied by RIP.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Functional Notes

Unlike the automatic summarization on classful network boundaries, only specific network advertisements are made by RIP using the **ip rip summary-address** command. This command is only effective if RIP version 2 is configured.

## Usage Examples

The following example enables manual summarization on the specified IP address:

(config)**#interface hdlc 1**
(config-hdlc 1)**#ip rip summary-address 10.10.123.0 255.255.255.0**

# ip route-cache

Use the **ip route-cache** command to enable fast-cache switching on the interface. Use the **no** form of this command to disable fast-cache switching and return to process switching mode.


*Using Network Address Translation (NAT) or the AOS firewall capabilities on an interface requires process switching mode (using the **no ip route-cache** command).*

## Syntax Description

No subcommands.

## Default Values

By default, fast-caching is enabled on all interfaces.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Functional Notes

Fast switching allows an IP interface to provide optimum performance when processing IP traffic.

## Usage Examples

The following example enables fast switching on the HDLC interface:

(config)**#interface hdlc 1**
(config-hdlc 1)**#ip route-cache**

# ip unnumbered *<interface>*

Use the **ip unnumbered** command to use the IP address assigned to the specified interface for all IP processing on the active interface. Use the **no** form of this command to remove the unnumbered configuration.

## Syntax Description

| | |
|---|---|
| *<interface>* | Specifies the interface that contains the IP address to use as the source address for all packets transmitted on this interface. Specify an interface in the format *<interface type [slot/port \| slot/port.sub-interface id \| interface id \| interface id.sub-interface id]>*. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **show ip unnumbered ?** for a list of valid interfaces. |

## Default Values

By default, all interfaces are configured to use a specified IP address (using the **ip address** command).

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |
| Release 11.1 | Command was expanded to include demand interface. |

## Functional Notes

If **ip unnumbered** is enabled on an interface, all IP traffic from the interface will use a source IP address taken from the specified interface. For example, specifying **ip unnumbered eth 0/1** while in the Frame Relay Sub-Interface Configuration mode configures the Frame Relay sub-interface to use the IP address assigned to the Ethernet interface for all IP processing. In addition, the AOS uses the specified interface information when sending route updates over the unnumbered interface.

## Usage Examples

The following example configures the HDLC interface to use the IP address assigned to the Ethernet interface 0/1:

(config)**#interface hdlc 1**
(config-hdlc 1)**#ip unnumbered eth 0/1**

# ip urlfilter *<name>*

Use the **ip urlfilter** command to apply a URL filter to the interface for all inbound or outbound traffic. Use the **no** form of this command to remove the URL filter from an interface. Variations of this command include:

**ip urlfilter** *<name>* **in**
**ip urlfilter** *<name>* **out**

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies the URL filter name to use on the interface. |
| **in** | Applies the filter to the inbound traffic. |
| **out** | Applies the filter to the outbound traffic. |

## Default Values

By default, there are no URL filters applied to any interfaces.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Functional Notes

The firewall must be enabled using the **ip firewall** command in order to use URL filters. The URL filter must be created by using the **ip urlfilter *<filtername>* http** command before applying it to the interface. See *ip urlfilter <name> http* on page 598 for more information on using this command.

## Usage Example

The following example performs URL filtering on all traffic entering through the HDLC interface and matches the URL filter named **MyFilter**:

(config)**#interface hdlc 1**
(config-hdlc 1)**#ip urlfilter MyFilter in**

# keepalive <*value*>

Use the **keepalive** command to enable the transmission of keepalive packets on the interface and specify the time interval in seconds between transmitted packets. Use the **no** form of this command to return to the default setting.

## Syntax Description

| | |
|---|---|
| <*value*> | Defines the time interval (in seconds) between transmitted keepalive packets. Valid range is 0 to 32,767 seconds. |

## Default Values

By default, the time interval between transmitted keepalive packets is 10 seconds.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Functional Notes

If three keepalive packets are sent to an interface with no response, the interface is considered down. To detect interface failures quickly, specify a smaller keepalive time.

## Usage Examples

The following example specifies a keepalive time of 5 seconds on the HDLC interface:

(config)**#interface hdlc 1**
(config-hdlc 1)**#keepalive 5**

# lldp receive

Use the **lldp receive** command to allow LLDP packets to be received on this interface. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, all interfaces are configured to send and receive LLDP packets.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Usage Examples

The following example configures the HDLC interface to receive LLDP packets:

```
(config)#interface hdlc 1
(config-hdlc 1)#lldp receive
```

# lldp send

Use the **lldp send** command to configure this interface to transmit LLDP packets or to control the types of information contained in the LLDP packets transmitted by this interface. Use the **no** form of these commands to disable these features. Variations of this command include:

**lldp send**
**lldp send management-address**
**lldp send port-description**
**lldp send system-capabilities**
**lldp send system-description**
**lldp send system-name**
**lldp send-and-receive**

## Syntax Description

| | |
|---|---|
| **management-address** | Enables transmission of management address information on this interface. |
| **port-description** | Enables transmission of port description information on this interface. |
| **system-capabilities** | Enables transmission of this device's system capabilities on this interface. |
| **system-description** | Enables transmission of this device's system description on this interface. |
| **system-name** | Enables transmission of this device's system name on this interface. |
| **and-receive** | Configures this interface to both transmit and receive LLDP packets. |

## Default Values

Be default, all interfaces are configured to transmit and receive LLDP packets of all types.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Functional Notes

Individual LLDP information can be enabled or disabled using the various forms of the **lldp send** command. For example, use the **lldp send and-receive** command to enable transmit and receive of all LLDP information. Then use the **no lldp send port-description** command to prevent LLDP from transmitting port description information.

## Usage Examples

The following example configures the HDLC interface to transmit LLDP packets containing all enabled information types:

(config)**#interface hdlc 1**
(config-hdlc 1)**#lldp send**

The following example configures the HDLC to transmit and receive LLDP packets containing all information types:

(config)**#interface hdlc 1**
(config-hdlc 1)**#lldp send-and-receive**

# max-reserved-bandwidth <*value*>

Use the **max-reserved-bandwidth** command specify the percentage of interface bandwidth reserved for use in user-defined (priority or class-based) queues. The remainder of the interface bandwidth is reserved for system critical traffic and is not available to user-defined queues. Use the **no** form of this command to restore the default values.


*Reserving a portion of the interface bandwidth for system critical traffic is necessary for proper operation. Specifying the entire interface bandwidth for use in user-defined queues can cause undesirable operation.*

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the maximum percentage of bandwidth to reserve for QoS. This setting is configured as a percentage of the total interface speed. Range is 1 to 100 percent. |

## Default Values

By default, **max-reserved-bandwidth** is set to 75 percent, which reserves 25 percent of the interface bandwidth for system critical traffic.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example specifies **85** percent of the bandwidth on the HDLC 1 be available for use in user-defined queues:

(config)**#interface hdlc 1**
(config-hdlc 1)**#max-reserved-bandwidth 85**

# media-gateway ip

Use the **media-gateway ip** command to associate an IP address source to use for RTP traffic. When configuring VoIP, RTP traffic needs an IP address to be associated with it. However, some interfaces allow "dynamic" configuration of IP addresses, and thus, this value could change periodically. Use the **no** form of these commands to disable these functions. Variations of this command include:

**media-gateway ip loopback** <*ip address*>
**media-gateway ip primary**
**media-gateway ip secondary** <*ip address*>

## Syntax Description

| | |
|---|---|
| **loopback** <*ip address*> | Use an IP address statically defined to a loopback interface. Helpful when using a single IP address across multiple WAN interfaces for RTP traffic. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| **primary** | Use the IP address that is configured as primary on this interface. Applies to static, DHCP, or negotiated addresses. |
| **secondary** <*ip address*> | Use the statically defined secondary IP address of this interface to be used for RTP traffic. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

## Default Values

By default, **media-gateway ip** is disabled.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example configures the unit to use the primary IP address for RTP traffic:

(config)**#interface hdlc 1**
(config-hdlc 1)**#media-gateway ip primary**

# mtu *<size>*

Use the **mtu** command to configure the maximum transmit unit (MTU) size for the active interface. Use the **no** form of this command to return to the default value.

## Syntax Description

| | | |
|---|---|---|
| *<size>* | Configures the window size for transmitted packets. The valid ranges for the various interfaces are listed below: | |
| | ATM interfaces | 64 to 1520 |
| | Demand interfaces | 64 to 1520 |
| | Ethernet interfaces | 64 to 1500 |
| | FDL interfaces | 64 to 256 |
| | HDLC interfaces | 64 to 1520 |
| | Loopback interfaces | 64 to 1500 |
| | Tunnel interfaces | 64 to 18,190 |
| | Virtual Frame Relay sub-interfaces | 64 to 1520 |
| | Virtual PPP interfaces | 64 to 1500 |

## Default Values

| | | |
|---|---|---|
| *<size>* | The default values for the various interfaces are listed below: | |
| | ATM interfaces | 1500 |
| | Demand interfaces | 1500 |
| | Ethernet interfaces | 1500 |
| | FDL interfaces | 256 |
| | HDLC interfaces | 1500 |
| | Loopback interfaces | 1500 |
| | Tunnel interfaces | 1500 |
| | Virtual Frame Relay sub-interfaces | 1500 |
| | Virtual PPP interfaces | 1500 |

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Functional Notes

OSPF will not become adjacent on links where the MTU sizes do not match. If router A and router B are exchanging hello packets but their MTU sizes do not match, they will never reach adjacency. This is by design and required by the RFC.

## Usage Examples

The following example specifies an MTU of 1200 on the HDLC interface:

(config)**#interface hdlc 1**
(config-hdlc 1)**#mtu 1200**

# qos-policy out *<name>*

Use the **qos-policy out** command to apply a previously-configured QoS map to outgoing packets on an interface. Use the **no** form of this command to remove the map from the interface.

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies the name of a previously-created QoS map (refer to *qos map <name> <number>* on page 643 for more information). |

## Default Values

No default value is necessary for this command.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Functional Notes

When a QoS policy is applied to an interface, it may be disabled if the interface bandwidth is not adequate to support the requested bandwidth on the map set. Once the bandwidth problem is resolved, the map will work again. The bandwidth will be rechecked on any of the following changes:

1. A priority or class-based entry is added to, deleted from, or changed in a QoS map set.
2. The interface bandwidth is changed by the **bandwidth** command on the interface.
3. A QoS policy is applied to an interface.
4. A cross-connect is created that includes an interface with a QoS policy.
5. The interface queuing method is changed to fair-queue to use weighted fair queuing.
6. The interface operational status changes.
7. The interface bandwidth changes for other reasons (e.g., when ADSL finishes training).

In order to prevent the map from being disabled in cases of temporary inadequate bandwidth (e.g., a single link goes down in a dual T1 multilink configuration where the map requests more than one T1's worth of bandwidth), the QoS map uses the maximum theoretical bandwidth on an interface, not the actual bandwidth at that time. This actually helps QoS keep higher priority class-based traffic working better than best-effort traffic when the bandwidth drops.

## Usage Examples

The following example applies the QoS map **VOICEMAP** to the HDLC interface:

(config)**#interface hdlc 1**
(config-hdlc 1)**#qos-policy out VOICEMAP**

# snmp trap link-status

Use the **snmp trap link-status** command to control the Simple Network Management Protocol (SNMP) variable ifLinkUpDownTrapEnable (RFC2863) to enable (or disable) the interface to send SNMP traps when there is an interface status change. Use the **no** form of this command to disable this trap.

## Syntax Description

No subcommands.

## Default Values

By default, the ifLinkUpDownTrapEnable OID is enabled for all interfaces except virtual Frame Relay interfaces.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Usage Examples

The following example disables the link-status trap on the HDLC interface:

(config)**#interface hdlc 1**
(config-hdlc 1)**#no snmp trap link-status**

# LOOPBACK INTERFACE CONFIGURATION COMMAND SET

To activate the Loopback Interface Configuration mode, enter the **interface loopback** command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#interface loopback 1
(config-loop 1)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:



All other commands for this command set are described in this section in alphabetical order.



# access-policy *<name>*

Use the **access-policy** command to assign a specified access policy to an interface. Access policies are applied to traffic entering an interface. Use the **no** form of this command to remove an access policy association. For more information on using access policies, refer to *ip policy-class <name>* on page 549.

*Configured access policies will only be active if the **ip firewall** command has been entered at the Global Configuration mode prompt to enable the AOS security features. All configuration parameters are valid, but no security data processing will be attempted unless the security features are enabled.*

## Syntax Description

| | |
|---|---|
| *<name>* | Identifies the configured access policy by alphanumeric descriptor (all access policy descriptors are case-sensitive). |

## Default Values

By default, there are no configured access policies associated with an interface.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include Ethernet sub-interfaces. |
| Release 6.1 | Command was expanded to include VLAN interfaces. |

## Functional Notes

To assign an access policy to an interface, enter the interface configuration mode for the desired interface and enter **access-policy** *<policy name>*.

## Usage Examples

The following example associates the access policy **Private** (to allow inbound traffic to the Web server) to the loopback interface 1:

Enable the AOS security features:
(config)**#ip firewall**

Associate the access policy with the loopback interface 1:

(config)**#interface loopback 1**
(config-loop 1)**#access-policy Private**

# bandwidth *<value>*

Use the **bandwidth** command to provide the bandwidth value of an interface to the higher-level protocols. This value is used in cost calculations. Use the **no** form of this command to restore the default values.

## Syntax Description

| | |
|---|---|
| *<value>* | Specifies bandwidth in kbps. Range is 1 to 4,294,967,295 kbps. |

## Default Values

To view default values, use the **show interfaces** command.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Functional Notes

The **bandwidth** command is an informational value that is communicated to the higher-level protocols to be used in cost calculations. This is a routing parameter only and does not affect the physical interface.

## Usage Examples

The following example sets bandwidth of the loopback interface to 10 Mbps:

(config)**#interface loopback 1**
(config-loop 1)**#bandwidth 10000**

# crypto map *<name>*

Use the **crypto map** command to associate crypto maps with the interface. Use the **no** form of this command to remove a crypto map from an interface.

> NOTE *When you apply a map to an interface, you are applying all crypto maps with the given map name. This allows you to apply multiple crypto maps if you have created maps which share the same name but have different map index numbers.*

> NOTE *For VPN configuration example scripts, refer to the technical support note **Configuring VPN** located on the ADTRAN OS System Documentation CD provided with your unit.*

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies the crypto map name that you wish to assign to the interface. |

## Default Values

By default, no crypto maps are assigned to an interface.

## Command History

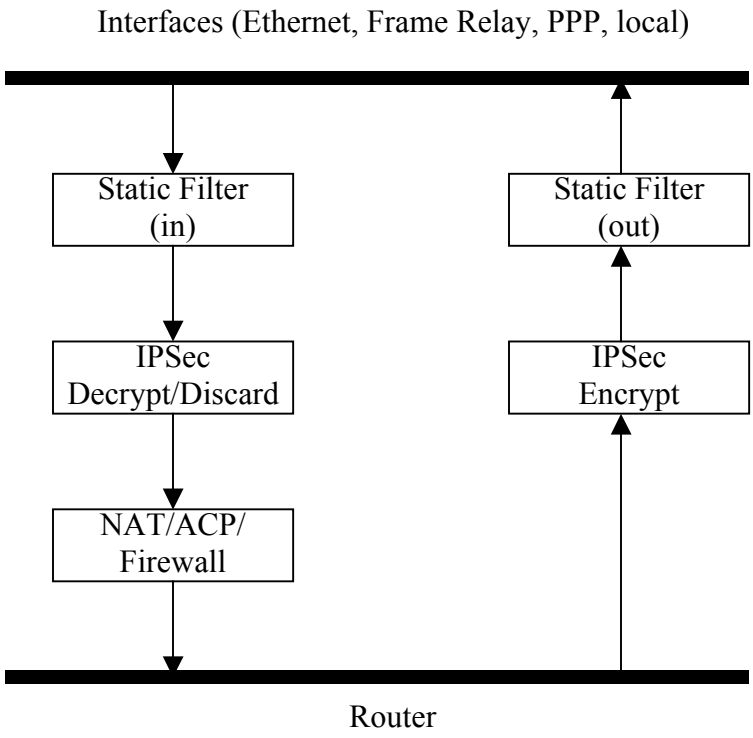
| | |
|---|---|
| Release 4.1 | Command was introduced. |

## Functional Notes

When configuring a system to use both the stateful inspection firewall and IKE negotiation for VPN, keep the following information in mind:

When defining the policy class and associated access-control lists (ACLs) that describe the behavior of the firewall, do not forget to include the traffic coming into the system over a VPN tunnel terminated by the system. The firewall should be set up with respect to the unencrypted traffic that is destined to be sent or received over the VPN tunnel. The following diagram represents typical AOS data-flow logic.




As shown in the diagram above, data coming into the product is first processed by the static filter associated with the interface on which the data is received. This access group is a true static filter and is available for use regardless of whether the firewall is enabled or disabled. Next (if the data is encrypted) it is sent to the IPSec engine for decryption. The decrypted data is then processed by the stateful inspection firewall. Therefore, given a terminating VPN tunnel, only unencrypted data is processed by the firewall.

The ACLs for a crypto map on an interface work in reverse logic to the ACLs for a policy class on an interface. When specifying the ACLs for a crypto map, the source information is the private local side, unencrypted source of the data. The destination information will be the far end, unencrypted destination of the data. However, ACLs for a policy class work in reverse. The source information for the ACL in a policy class is the far end. The destination information is the local-side.

## Usage Examples

The following example applies all crypto maps with the name **MyMap** to the loopback interface:

(config)**#interface loopback 1**
(config-loop 1)**#crypto map MyMap**

# dynamic-dns

Use the **dynamic-dns** command to configure Dynamic DNS service provided by Dynamic Network Services, Inc. (www.dyndns.org). Use the **no** versions of these commands to disable these features. Variations of this command include:

**dynamic-dns custom** <*hostname*> <*minutes*>
**dynamic-dns dyndns** <*hostname*> <*username*> <*password*>
**dynamic-dns dyndns-custom** <*hostname*> <*username*> <*password*>
**dynamic-dns dyndns-static** <*hostname*> <*username*> <*password*>

## Syntax Description

| | |
|---|---|
| <*hostname*> | Specifies the hostname for the server that updates the Dynamic Domain Name Server (DNS). |
| <*minutes*> | Specifies the intervals in minutes to update the server with information (updates also occur when the interface's IP address changes regardless of the update intervals).<br>Refer to *Functional Notes* below for additional argument descriptions. |

## Default Values

No default is necessary for this command.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |
| Release 12.1 | Command was expanded. |

## Functional Notes

**custom** - Constanttime.com's Custom Dynamic DNS$^{SM}$ service allows you complete access and management control over your domain name regardless of where you purchased/registered it. This allows to manage IP address mappings (A records), domain aliases (CNAME records) and mail servers (MX records).

**dyndns** - The Dynamic DNS$^{SM}$ offered by Dynamic Network Services Inc., (DynDNS.org) allows you to alias a dynamic IP address to a static hostname in various domains. This allows your unit to be more easily accessed from various locations on the Internet. This service is provided for up to five hostnames.

**dyndns-custom -** DynDNS.org's Custom DNS$^{SM}$ service provides a full DNS solution, giving you complete control over an entire domain name. A web-based interface provides two levels of control over your domain, catering to average or advanced users. Five globally redundant DNS servers ensure that your domain will always resolve.

A choice of two interfaces is available. The basic interface is designed for most users. It comes preconfigured for the most common configuration and allows for easy creation of most common record types. The advanced interface is designed for system administrators with a solid DNS background, and provides layout and functionality similar to a BIND zone file allowing for the creation of nearly any record type.

Custom DNS$^{SM}$ can be used with both static and dynamic IPs and has the same automatic update capability through Custom DNS-aware clients as Dynamic DNS.

**dyndns-static -** The Static DNS service is similar to DynDNS.org's Dynamic DNS$^{SM}$ service in that it allows a hostname such as yourname.dyndns.org to point to your IP address. Unlike a Dynamic DNS host, a Static DNS host does not expire after 35 days without updates, but updates take longer to propagate though the DNS system. This service is provided for up to five hostnames.

If your IP address does not change often or at all but you still want an easy name to remember it by (without having to purchase your own domain name), Static DNS service is ideal for you.

If you would like to use your own domain name (such as yourname.com), you need Custom DNS service which also provides full dynamic and static IP address support.

**Usage Examples**

The following example sets the dynamic-dns to **dyndns-custom** with hostname **host**, username **user**, and password **pass**:

(config)**#interface loopback 1**
(config-loop 1)**#dynamic-dns dyndns-custom host user pass**

# ip access-group *<name>*

Use the **ip access-group** command to create an access control list to be used for packets transmitted on or received from the specified interface. Use the **no** form of this command to disable this type of control. Variations of this command include:

**ip access-group** <*name*> **in**
**ip access-group** <*name*> **out**

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies IP access control list name. |
| **in** | Enables access control on packets received on the specified interface. |
| **out** | Enables access control on packets transmitted on the specified interface. |

## Default Values

By default, these commands are disabled.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Functional Notes

When this command is enabled, the IP destination address of each packet must be validated before being passed through. If the packet is not acceptable per these settings, it is dropped.

## Usage Examples

The following example sets up the router to allow only Telnet traffic into the loopback interface:

(config)**#ip access-list extended TelnetOnly**
(config-ext-nacl)**#permit tcp any any eq telnet**
(config-ext-nacl)**#interface loopback 1**
(config-loop 1)**#ip access-group TelnetOnly in**

# ip address *<ip address> <subnet mask>*

Use the **ip address** command to define an IP address on the specified interface. Use the optional **secondary** keyword to define a secondary IP address. Use the **no** form of this command to remove a configured IP address. Variations of this command include:

**ip address** *<ip address> <subnet mask>*
**ip address** *<ip address> <subnet mask>* **secondary**

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies a valid IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| *<subnet mask>* | Specifies the subnet mask that corresponds to a range of IP addresses (network) or a specific host. Subnet masks can be expressed in dotted decimal notation (for example, **255.255.255.0**) or as a prefix length (for example, **/24**). |
| **secondary** | Optional. Configures a secondary IP address for the specified interface. |

## Default Values

By default, there are no assigned IP addresses.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 2.1 | Added **ip address dhcp** for DHCP client support. |

## Functional Notes

Use secondary IP addresses to allow dual subnets on a single interface (when you need more IP addresses than the primary subnet can provide). When using secondary IP addresses, avoid routing loops by verifying that all devices on the network segment are configured with secondary IP addresses on the secondary subnet.

## Usage Examples

The following example configures a secondary IP address of **192.22.72.101/30**:

(config)**#interface loopback 1**
(config-loop 1)**#ip address 192.22.72.101 255.255.255.252 secondary**

# ip ffe

Use the **ip ffe** command to enable the FastFlow Engine (FFE) on this interface with the default number of entries. Use the **no** form of these commands to disable the FFE. Variations of this command include:

**ip ffe max-entries** <*entries*>

*Issuing this command will cause all FFE entries on this interface to be cleared.*

## Syntax Description

| | |
|---|---|
| **max-entries** <*entries*> | Specifies the maximum number of entries stored in the flow table. Valid range is from 1 to 8192. |

## Default Values

By default, the FFE is disabled. The default number of **max-entries** is 4096.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Functional Notes

The FFE can be used to help reduce routing overhead and thus reduce overall routing times. Routing times are reduced by the creation of a flow table on the ingress interface. The maximum number of entries that can be stored in the flow table at any one time may be specified by using the **max-entries** subcommand. When features such as firewall, VPN, and static filters are enabled, routing time reduction may not be realized.

## Usage Examples

The following example enables the FFE and sets the maximum number of entries in the flow table to 50:

(config)**#interface loopback 1**
(config-loop 1)**#ip ffe max-entries 50**

## Technology Review

The FastFlow system goal is to increase IP packet throughput by moving as much of the packet processing into the engine as possible. Packets are classified into flows based upon the IP protocol (TCP, UDP, ICMP, etc.), the source and destination IP addresses, IP TOS, and the protocol-specific information such as the source and destination port numbers. Flows are defined as the unidirectional representation of a conversation between two IP hosts. Each ingress interface keeps its own flow table, a collection of flow entries.

The first packet in a flow that is forwarded through the unit will build a flow entry. When a flow entry is looked up but no entry is found, a FastFlowBuilder object is allocated and attached to the packet. As the packet passes through the various processing layers, each subsystem will add processing to the FastFlowBuilder. When packet is about to be forwarded out of the egress interface, the FastFlowBuilder will be finalized. That is, the flow entry being built will be checked for completeness and committed to the flow table on the ingress interface. Subsequent flow matches can then bypass the normal processing layers.

# ip helper-address *<ip address>*

Use the **ip helper-address** command to configure the AOS to forward User Datagram Protocol (UDP) broadcast packets received on the interface. Use the **no** form of this command to disable forwarding packets.

> NOTE *The **ip helper** command must be used in conjunction with the **ip forward-protocol** command to configure the AOS to forward UDP broadcast packets. Refer to ip forward-protocol udp <value> on page 532 for more information.*

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies the destination IP address for the forwarded UDP packets. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

## Default Values

By default, broadcast UDP packets are not forwarded.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

When used in conjunction with the **ip forward-protocol** command, the **ip helper-address** feature allows you to customize which broadcast packets are forwarded.

To implement the helper address feature, assign a helper-address(es) (specifying the device that needs to receive the broadcast traffic) to the interface closest to the host that transmits the broadcast packets. When broadcast packets (of the specified type forwarded using the **ip forward-protocol** command) are received on the interface, they will be forwarded to the device that needs the information.

Only packets meeting the following criteria are considered eligible by the **ip helper-address** feature:

1. The packet IP protocol is User Datagram Protocol (UDP).

2. Any UDP port specified using the **ip forward-protocol** command.

3. The media access control (MAC) address of the frame is an all-ones broadcast address (ffff.ffff.ffff).

4. The destination IP address is broadcast defined by all ones (255.255.255.255) or a subnet broadcast (for example, 192.33.4.251 for the 192.33.4.248/30 subnet).

## Usage Examples

The following example forwards all DNS broadcast traffic to the DNS server with IP address **192.33.5.99**:

(config)**#ip forward-protocol udp domain**
(config)**#interface loopback 1**
(config-loop 1)**#ip helper-address 192.33.5.99**

# ip igmp

Use the **ip igmp** command to configure multicasting-related functions for the interface. Variations of this command include:

**ip igmp immediate-leave**
**ip igmp last-member-query-interval** <*milliseconds*>
**ip igmp querier-timeout** <*seconds*>
**ip igmp query-interval** <*seconds*>
**ip igmp query-max-response-time** <*seconds*>
**ip igmp static-group** <*address*>
**ip igmp version [1 | 2]**

## Syntax Description

| | |
|---|---|
| **immediate-leave** | Specifies that if only one host (or IGMP snooping switch) is connected to the interface, when a leave is received, multicast of that group is immediately terminated as opposed to sending a group query and timing out the group if no device responds. Works in conjunction with **ip igmp last-member-query-interval**. Applies to all groups when configured. Use the **no** form of this command to disable the **immediate-leave** feature. |
| **last-member-query-interval** <*milliseconds*> | Controls the timeout (in milliseconds) used to detect whether any group receivers remain on an interface after a receiver leaves a group. If a receiver sends a leave-group message (IGMP Version 2), the router sends a group-specific query on that interface. After twice the time specified by this command plus as much as one second longer, if no receiver responds, the router removes that interface from the group and stops sending that group's multicast packets to the interface. Range is 100 to 65,535 ms. Use the **no** form of this command to return to the default setting. |
| **querier-timeout** <*seconds*> | Specifies the interval (in seconds) that the router waits after the current querier's last query before it takes over as querier (IGMP V2). Range is 60 to 300 seconds. Use the **no** form of this command to return to the default setting. |
| **query-interval** <*seconds*> | Specifies the interval (in seconds) at which IGMP queries are sent on an interface. Host query messages are addressed to the all-hosts multicast group with an IP TTL of 1. The router uses queries to detect whether multicast group members are on the interface and to select an IGMP designated router for the attached segment (if more than one multicast router exists). Only the designated router for the segment sends queries. For IGMP V2, the designated router is the router with the lowest IP address on the segment. Range is 0 to 65,535 seconds. Use the **no** form of this command to return to the default setting. |
| **query-max-response-time** <*seconds*> | Specifies the maximum response time (in seconds) advertised by this interface in queries when using IGMP V2. Hosts are allowed a random time within this period to respond, reducing response bursts. Use the **no** form of this command to return to the default setting. |

### Syntax Description

| | |
|---|---|
| **static-group** <*address*> | Configures the router's interface to be a statically-connected member of the specified group. Packets received on the correct RPF interface are forwarded to this interface regardless of whether any receivers have joined the specified group using IGMP. Use the **no** form of this command to remove a configured static group. |
| **version [1 \| 2]** | Sets the interface's IGMP version. Use the **no** form of this command to return to the default setting. |

### Default Values

| | |
|---|---|
| **ip igmp immediate-leave** | No default |
| **ip igmp last-member-query-interval** | 1000 milliseconds |
| **ip igmp querier-timeout** | 2x the **query-interval** value |
| **ip igmp query-interval** | 60 seconds |
| **ip igmp query-max-response-time** | 10 seconds |
| **ip igmp static-group** | No default |
| **ip igmp version** | Version 1 |

### Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |

### Usage Examples

The following example sets the query message interval on the interface to 200 milliseconds:

```
(config)#interface loopback 1
(config-loop 1)#ip igmp last-member-query-interval 200
```

# ip mcast-stub downstream

Use the **ip mcast-stub downstream** command to enable multicast forwarding and IGMP (router mode) on an interface and place it in multicast stub downstream mode. Use the **no** form of this command to disable.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |

## Functional Notes

This command is used in IP multicast stub applications in conjunction with the **ip mcast-stub helper-address** and **ip mcast-stub upstream** commands. Downstream interfaces connect to segments with multicast hosts. Multiple interfaces may be configured in downstream mode; however, interfaces connecting to the multicast network (upstream) should not be configured in downstream mode. Interfaces configured as downstream should have the lowest IP address of all IGMP-capable routers on the connected segment in order to be selected as the designated router and ensure proper forwarding. Refer to *ip mcast-stub helper-address <ip address>* on page 546 and *ip mcast-stub upstream* on page 1333 for more information.

## Usage Examples

The following example enables multicast forwarding and IGMP on the interface:

```
(config)#interface loopback 1
(config-loop 1)#ip mcast-stub downstream
```

# ip mcast-stub fixed

Use the **ip mcast-stub fixed** command to allow forwarding of multicast traffic on a selected interface after enabling multicast routing. Use the **no** form of this command to disable this mode.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Functional Notes

Multicast routing must be enabled prior to setting **ip mcast-stub fixed** on the selected interface. Also, use the **ip igmp static-group** <*A.B.C.D*> command to receive multicast traffic without host-initiated Internet Group Management Protocol (IGMP) activity on the selected interface. Otherwise, all host-initiated IGMP transactions will enter multicast routes on the router's interface involved with IGMP activities.

### Usage Examples

The following example enables multicast traffic forwarding and IGMP on the interface:

(config)**#interface loopback 1**
(config-loop 1)**#ip mcast-stub fixed**

# ip mcast-stub helper-enable

Use the **ip mcast-stub helper-enable** command to assign the **ip mcast-stub helper-address** as the IGMP proxy. Use the **no** form of this command to disable.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |
| Release 10.1 | Command was expanded to include loopback interfaces. |

## Functional Notes

This command is used in IP multicast stub applications in conjunction with the **ip mcast-stub helper-address**, **ip mcast-stub upstream**, and **ip mcast-stub downstream** commands. When enabled, the interface becomes a helper forwarding interface. The IGMP host function is dynamically enabled and the interface becomes the active upstream interface, enabling the unit to perform as an IGMP proxy. Refer to *ip mcast-stub helper-address <ip address>* on page 546, *ip mcast-stub downstream* on page 1330, and *ip mcast-stub upstream* on page 1333 for more information.

## Usage Examples

The following example sets the helper address as the IGMP proxy:

(config)**#interface loopback 1**
(config-loop 1)**#ip mcast-stub helper-enable**

# ip mcast-stub upstream

Use the **ip mcast-stub upstream** command to enable multicast forwarding on an interface and place it in multicast stub upstream mode. Use the **no** form of this command to disable.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |

## Functional Notes

This command is used in IP multicast stub applications in conjunction with the **ip mcast-stub helper-address** and **ip mcast-stub downstream** commands. When enabled, the interface becomes a candidate to be a helper forwarding interface. If chosen as the best path toward the helper address by the router's unicast route table, the IGMP host function is dynamically enabled and the interface becomes the active upstream interface, enabling the router to perform as an IGMP proxy. Though multiple interfaces may be candidates, no more than one interface will actively serve as the helper forwarding interface. Refer to *ip mcast-stub helper-address <ip address>* on page 546 and *ip mcast-stub downstream* on page 1330 for more information.

## Usage Examples

The following example enables multicast forwarding on the interface:

(config)**#interface loopback 1**
(config-loop 1)**#ip mcast-stub upstream**

# ip ospf

Use the **ip ospf** command to customize OSPF settings (if needed). Use the **no** form of these commands to return to the default settings. Variations of this command include:

**ip ospf authentication-key** <*password*>
**ip ospf cost** <*value*>
**ip ospf dead-interval** <*seconds*>
**ip ospf hello-interval** <*seconds*>
**ip ospf message-digest-key [1 | 2] md5** <*key*>
**ip ospf priority** <*value*>
**ip ospf retransmit-interval** <*seconds*>
**ip ospf transmit-delay** <*seconds*>

## Syntax Description

| | |
|---|---|
| **authentication-key** <*password*> | Assigns a simple-text authentication password to be used by other routers using the OSPF simple password authentication. |
| **cost** <*value*> | Specifies the OSPF cost of sending a packet on the interface. This value overrides any computed cost value. Range is 1 to 65,535. |
| **dead-interval** <*seconds*> | Sets the maximum interval (in seconds) allowed between hello packets. If the maximum is exceeded, neighboring devices will determine that the device is down. Range is 0 to 32,767 seconds. |
| **hello-interval** <*seconds*> | Specifies the interval (in seconds) between hello packets sent on the interface. Range is 0 to 32,767 seconds. |
| **message-digest-key [1 \| 2] md5** <*key*> | Configures OSPF Message Digest 5 (MD5) authentication (16-byte max) keys. |
| **priority** <*value*> | Sets the OSPF priority. The value set in this field helps determine the designated router for this network. Range is 0 to 255. |
| **retransmit-interval** <*seconds*> | Specifies the interval (in seconds) between link-state advertisements (LSAs). Range is 0 to 32,767 seconds. |
| **transmit-delay** <*seconds*> | Sets the estimated time (in seconds) required to send an LSA on the interface. Range is 0 to 32,767 seconds. |

## Default Values

| | |
|---|---|
| **retransmit-interval** <*seconds*> | 5 seconds |
| **transmit-delay** <*seconds*> | 1 second |
| **hello-interval** <*seconds*> | 10 seconds: Ethernet, point-to-point, Frame Relay, and PPP |
| **dead-interval** <*seconds*> | 40 seconds |

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Usage Examples

The following example sets the maximum number of seconds allowed between hello packets to 25,000:

(config)**#interface loopback 1**
(config-loop 1)**#ip ospf dead-interval 25000**

# ip ospf authentication

Use the **ip ospf authentication** command to authenticate an interface that is performing OSPF authentication. Use the **no** form of this command to return to the default setting. Variations of this command include:

**ip ospf authentication message-digest**
**ip ospf authentication null**

## Syntax Description

| | |
|---|---|
| **message-digest** | Optional. Specifies the message-digest authentication type. |
| **null** | Optional. Specifies for no authentication to be used. |

## Default Values

By default, this is set to **null** (meaning no authentication is used).

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Usage Examples

The following example specifies that no authentication will be used on the loopback interface:

(config)**#interface loopback 1**
(config-loop 1)**#ip ospf authentication null**

# ip ospf network

Use the **ip ospf network** command to specify the type of network on this interface. Use the **no** form of this command to return to the default setting. Variations of this command include:

**ip ospf network broadcast**
**ip ospf network point-to-point**

## Syntax Description

| | |
|---|---|
| **broadcast** | Sets the network type for broadcast. |
| **point-to-point** | Sets the network type for point-to-point. |

## Default Values

By default, Ethernet defaults to broadcast. PPP and Frame Relay default to point-to-point.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Functional Notes

A point-to-point network will not elect designated routers.

## Usage Examples

The following example designates a broadcast network type:

(config)**#interface loopback 1**
(config-loop 1)**#ip ospf network broadcast**

# ip pim sparse-mode

Use the **ip pim sparse-mode** command to enable protocol-independent multicast (PIM) sparse mode for this interface. Use the **no** form of this command to disable PIM sparse mode.

## Syntax Description

No subcommands.

## Default Values

By default, PIM sparse mode for this interface is disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Functional Notes

PIM Sparse Mode is a multicast routing protocol that makes use of the unicast forwarding table. It builds unidirectional shared trees rooted at a Rendezvous Point (RP) for a multicast group or a shortest-path tree rooted at a specific source for a multicast group.

## Usage Examples

The following example enables PIM sparse mode on the interface:

(config)**#interface loopback 1**
(config-loop 1)**#ip pim sparse-mode**

# ip pim-sparse dr-priority <*value*>

Use the **ip pim-sparse dr-priority** command to specify the priority for the designated router (DR). This command modifies the routers priority in the DR election process. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the priority of this interface (to be used when determining the DR). Valid range is 1 to 4,294,967,295. |

## Default Values

By default, the priority of all PIM interfaces is 1.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Functional Notes

Interfaces advertise their configured priority values in the hello messages transmitted on the interface. Routers use the priority values to determine the appropriate DR. The router on the network segment with the highest priority is selected as the DR. If a hello message is received on the interface from a router on the network segment and it does not contain a priority, the entire network segment defaults to DR selection based on IP addresses instead of priority. In this instance, the DR is selected as the router on the network segment that has the highest IP address. AOS will always include a priority in all transmitted hello messages. If no priority is specifically designated by the user, the priority is set as the default of 1.

## Usage Examples

The following example specifies a priority of **100** on the Loopback 1 interface:

(config)**#interface loopback 1**
(config-loop 1)**#ip pim-sparse dr-priority 100**

# ip pim-sparse hello-timer <*value*>

Use the **ip pim-sparse hello-timer** command to specify protocol-independent multicast (PIM) sparse hello timer period. This is the time interval at which periodic hellos are sent out on all interfaces of a PIM-capable router. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the interval (in seconds) at which periodic hellos are sent out of the interface. Valid range is 10 to 3600 seconds. |

## Default Values

By default, the hellos are transmitted on PIM interfaces every 60 seconds.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Functional Notes

Hello messages are used to inform neighbors of a router's presence. Hello messages normally generate a small amount of traffic on an interface. Setting the **hello-timer** to a small interval increases the number of hellos sent (thus increasing the amount of traffic). Set the **hello-timer** to a reasonable value, taking into consideration the bandwidth available on the interface.

## Usage Examples

The following example specifies hellos be sent on the Loopback 1 interface every **3600** seconds:

(config)**#interface loopback 1**
(config-loop 1)**#ip pim-sparse hello-timer 3600**

# ip pim-sparse nbr-timeout <*value*>

Use the **ip pim-sparse nbr-timeout** command to specify protocol-independent multicast (PIM) sparse neighbor timeout. This is the time interval after which a PIM-capable router will consider a neighbor not present. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the time interval in seconds after which a neighbor is considered not present. Valid range is 30 to 10,800 seconds. |

## Default Values

By default, the PIM sparse neighbor timeout is set to 105 seconds.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example sets the nbr-timeout to 300 seconds:

(config)**#interface loopback 1**
(config-loop 1)**#ip pim-sparse nbr-timeout 300**

# ip pim-sparse override-interval <*value*>

Use the **ip pim-sparse override-interval** command to specify the protocol-independent multicast (PIM) sparse join/prune override interval. This delay interval is the period after a join/prune that another router on the LAN may override a join/prune. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the delay time in milliseconds. Valid range is 0 to 65,535 milliseconds. |

## Default Values

By default, the override interval is set to 2500 milliseconds.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example sets the override interval to 3000 milliseconds:

(config)**#interface loopback 1**
(config-loop 1)**#ip pim-sparse override-interval 3000**

# ip pim-sparse propagation-delay <*value*>

Use the **ip pim-sparse propagation-delay** command to specify the expected propagation delay for join/prune messages. Set the propagation delay (in milliseconds) to estimate the amount of delay found in the local link. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the expected propagation delay in the local link in milliseconds. Valid range is 0 to 32,767 milliseconds. |

## Default Values

By default, the propagation delay is set to 500 milliseconds.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example sets the propagation delay to **300** milliseconds:

(config)**#interface loopback 1**
(config-loop 1)**#ip pim-sparse propagation-delay 300**

# ip policy route-map *<name>*

Use the **ip policy route-map** command to assign a policy route-map to this interface. Use the **no** form of this command to remove the route-map policy.

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies the name of the policy route map to assign to this interface. |

## Default Values

By default, no policy route map is assigned to this interface.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example assigns the policy route map **policy1** to the interface:

(config)**#interface loopback 1**
(config-loop 1)**#ip policy route-map policy1**

# ip proxy-arp

Use the **ip proxy-arp** command to enable proxy Address Resolution Protocol (ARP) on the interface. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, proxy arp is enabled.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

In general, the principle of proxy ARP allows a router to insert its IP address in the source IP address field of a packet (if the packet is from a host on one of its subnetworks). This allows hosts to reach devices on other subnetworks without implementing routing or specifying a default gateway.

If proxy-arp is enabled, the AOS will respond to all proxy-arp requests with its specified MAC address and forward packets accordingly.

Enabling proxy-arp on an interface may introduce unnecessary ARP traffic on the network.

## Usage Examples

The following enables proxy ARP on the loopback interface:

(config)**#interface loopback 1**
(config-loop 1)**#ip proxy-arp**

# ip rip receive version

Use the **ip rip receive version** command to configure the RIP version the unit accepts in all RIP packets received on the interface. Use the **no** form of this command to return to the default setting. Variations of this command include:

**ip rip receive version 1**
**ip rip receive version 2**

## Syntax Description

| | |
|---|---|
| **1** | Accepts only received RIP version 1 packets on the interface. |
| **2** | Accepts only received RIP version 2 packets on the interface. |

## Default Values

By default, all interfaces implement RIP version 1 (the default value for the **version** command).

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

Use the **ip rip receive version** to specify a RIP version that overrides the **version** (in the Router RIP) configuration.

The AOS only accepts one version (either 1 or 2) on a given interface.

## Usage Examples

The following example configures the loopback interface to accept only RIP version 2 packets:

(config)**#interface loopback 1**
(config-loop 1)**#ip rip receive version 2**

# ip rip send version

Use the **ip rip send version** command to configure the RIP version the unit sends in all RIP packets transmitted on the interface. Use the **no** form of this command to return to the default setting. Variations of this command include:

**ip rip send version 1**
**ip rip send version 2**

## Syntax Description

| | |
|---|---|
| **1** | Transmits only RIP version 1 packets on the interface. |
| **2** | Transmits only RIP version 2 packets on the interface. |

## Default Values

By default, all interfaces transmit RIP version 1 (the default value for the **version** command).

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

Use the **ip rip send version** to specify a RIP version that overrides the **version** (in the Router RIP) configuration.

The AOS only transmits one version (either 1 or 2) on a given interface.

## Usage Examples

The following example configures the loopback interface to transmit only RIP version 2 packets:

(config)**#interface loopback 1**
(config-loop 1)**#ip rip send version 2**

# ip rip summary-address *<ip address> <subnet mask>*

Use the **ip rip summary-address** command to manually summarize the routes Routing Information Protocol (RIP) will advertise and send out a specified interface. Use the **no** form of this command to disable this mode.

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies the summarized network IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.0**). |
| *<subnet mask>* | Specifies the subnet mask that corresponds to the range of IP addresses (network). Subnet masks can be expressed in dotted decimal notation (for example, **255.255.255.0**) or as a prefix length (for example, **/24**). |

## Default Values

By default, no manual summarization is applied by RIP.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Functional Notes

Unlike the automatic summarization on classful network boundaries, only specific network advertisements are made by RIP using the **ip rip summary-address** command. This command is only effective if RIP version 2 is configured.

## Usage Examples

The following example enables manual summarization on the specified IP address:

(config)**#interface eth 0/1**
(config-eth 0/1)**#ip rip summary-address 10.10.123.0 255.255.255.0**

# ip route-cache

Use the **ip route-cache** command to enable fast-cache switching on the interface. Use the **no** form of this command to disable fast-cache switching and return to process switching mode.

*Using Network Address Translation (NAT) or the AOS firewall capabilities on an interface requires process switching mode (using the **no ip route-cache** command).*

## Syntax Description

No subcommands.

## Default Values

By default, fast-cache switching is enabled on all Ethernet and virtual Frame Relay sub-interfaces. IP route-cache is enabled for all virtual PPP interfaces.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |

## Functional Notes

Fast switching allows an IP interface to provide optimum performance when processing IP traffic.

## Usage Examples

The following example enables fast switching on the loopback interface:

(config)**#interface loopback 1**
(config-loop 1)**#ip route-cache**

# ip unnumbered *<interface>*

Use the **ip unnumbered** command to use the IP address assigned to the specified interface for all IP processing on the active interface. Use the **no** form of this command to remove the unnumbered configuration.

## Syntax Description

| | |
|---|---|
| *<interface>* | Specifies an interface in the format *<interface type [slot/port \| slot/port.sub-interface id \| interface id \| interface id.sub-interface id]>*. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **ip unnumbered ?** for a complete list of valid interfaces. |

## Default Values

By default, all interfaces are configured to use a specified IP address (using the **ip address** command).

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 11.1 | Command was expanded to include demand interfaces. |

## Functional Notes

If **ip unnumbered** is enabled on an interface, all IP traffic from the interface will use a source IP address taken from the specified interface. For example, specifying **ip unnumbered ppp 1** while in the Ethernet Interface Configuration mode configures the Ethernet interface to use the IP address assigned to the PPP interface for all IP processing. In addition, the AOS uses the specified interface information when sending route updates over the unnumbered interface.

## Usage Examples

The following example configures the loopback interface (labeled **loop 1**) to use the IP address assigned to the PPP interface (**ppp 1**):

(config)**#interface loopback 1**
(config-loop 1)**#ip unnumbered ppp 1**

# ip urlfilter *<name>*

Use the **ip urlfilter** command to apply a URL filter to the interface for all inbound or outbound traffic. Use the **no** form of this command to remove the URL filter from an interface. Variations of this command include:

**ip urlfilter** *<name>* **in**
**ip urlfilter** *<name>* **out**

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies the URL filter name to use on the interface. |
| **in** | Applies the filter to the inbound traffic. |
| **out** | Applies the filter to the outbound traffic. |

## Default Values

By default, there are no URL filters applied to any interfaces.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Functional Notes

The firewall must be enabled using the **ip firewall** command in order to use URL filters. The URL filter must be created by using the **ip urlfilter** *<filtername>* **http** command before applying it to the interface. See *ip urlfilter <name> http* on page 598 for more information on using this command.

## Usage Example

The following example performs URL filtering on all traffic entering through the loopback interface (labeled **loop 1**) and matches the URL filter named **MyFilter**:

(config)**#interface loopback 1**
(config-loop 1)**#ip urlfilter MyFilter in**

# mtu *<size>*

Use the **mtu** command to configure the maximum transmit unit size for the active interface. Use the **no** form of this command to return to the default value.

## Syntax Description

| | | |
|---|---|---|
| *<size>* | Configures the window size for transmitted packets. The valid ranges for the various interfaces are listed below: | |
| | ATM interfaces | 64 to 1520 |
| | Demand interfaces | 64 to 1520 |
| | Ethernet interfaces | 64 to 1500 |
| | FDL interfaces | 64 to 256 |
| | HDLC interfaces | 64 to 1520 |
| | Loopback interfaces | 64 to 1500 |
| | Tunnel interfaces | 64 to 18,190 |
| | Virtual Frame Relay sub-interfaces | 64 to 1520 |
| | Virtual PPP interfaces | 64 to 1500 |

## Default Values

| | | |
|---|---|---|
| *<size>* | The default values for the various interfaces are listed below: | |
| | ATM interfaces | 1500 |
| | Demand interfaces | 1500 |
| | Ethernet interfaces | 1500 |
| | FDL interfaces | 256 |
| | HDLC interfaces | 1500 |
| | Loopback interfaces | 1500 |
| | Tunnel interfaces | 1500 |
| | Virtual Frame Relay sub-interfaces | 1500 |
| | Virtual PPP interfaces | 1500 |

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

OSPF will not become adjacent on links where the MTU sizes do not match. If router A and router B are exchanging hello packets but their MTU sizes do not match, they will never reach adjacency. This is by design and required by the RFC.

## Usage Examples

The following example specifies an MTU of 1200 on the loopback interface:

(config)**#interface loopback 1**
(config-loop 1)**#mtu 1200**

# snmp trap

Use the **snmp trap** command to enable all supported Simple Network Management Protocol (SNMP) traps on the interface. Use the **no** form of this command to disable this trap.

## Syntax Description

No subcommands.

## Default Values

By default, all interfaces (except virtual Frame Relay interfaces and sub-interfaces) have SNMP traps enabled.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 3.1 | Command was extended to the SHDSL interface. |
| Release 5.1 | Command was expanded to include Ethernet sub-interfaces and Gigabit Ethernet interfaces. |

## Usage Examples

The following example enables SNMP capability on the Ethernet interface:

(config)**#interface eth 0/1**
(config-eth 0/1)**#snmp trap**

# snmp trap link-status

Use the **snmp trap link-status** command to control the Simple Network Management Protocol (SNMP) variable ifLinkUpDownTrapEnable (RFC2863) to enable (or disable) the interface to send SNMP traps when there is an interface status change. Use the **no** form of this command to disable this trap.

## Syntax Description

No subcommands.

## Default Values

By default, the ifLinkUpDownTrapEnable OID is enabled for all interfaces except virtual Frame Relay interfaces.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 3.1 | Command was extended to the SHDSL interface. |
| Release 5.1 | Command was expanded to include Gigabit Ethernet, port channel, VLAN, E1, and G.703 interfaces. |

## Functional Notes

The **snmp trap link-status** command is used to control the RFC2863 ifLinkUpDownTrapEnable OID (OID number 1.3.6.1.2.1.31.1.1.1.14.0).

## Usage Examples

The following example disables the link-status trap on the loopback interface:

(config)**#interface loopback 1**
(config-loop 1)**#no snmp trap link-status**

# PORT CHANNEL INTERFACE CONFIG COMMAND SET

To activate the Port Channel Interface Configuration mode, enter the **interface port-channel** command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#interface port-channel 1
(config-p-chan 1)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:



All other commands for this command set are described in this section in alphabetical order.



# arp arpa

Use the **arp arpa** command to set ARPA as the standard address resolution protocol (ARP) on this interface. Use the **no** form of this command to return to the default setting.

## Syntax Description

No subcommands.

## Default Values

The default for this command is **arpa**.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following example enables standard ARP for the port channel interface:

(config)**#interface port-channel 1**
(config-p-chan 1)**#arp arpa**

# lldp receive

Use the **lldp receive** command to allow LLDP packets to be received on this interface. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, all interfaces are configured to send and receive LLDP packets.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Usage Examples

The following example configures the port channel interface to receive LLDP packets:

(config)**#interface port-channel 1**
(config-p-chan 1)**#lldp receive**

# lldp send

Use the **lldp send** command to configure this interface to transmit LLDP packets or to control the types of information contained in the LLDP packets transmitted by this interface. Use the **no** form of these command to disable these features. Variations of this command include:

**lldp send management-address**
**lldp send port-description**
**lldp send system-capabilities**
**lldp send system-description**
**lldp send system-name**
**lldp send-and-receive**

## Syntax Description

| | |
|---|---|
| **management-address** | Enables transmission of management address information on this interface. |
| **port-description** | Enables transmission of port description information on this interface. |
| **system-capabilities** | Enables transmission of this device's system capabilities on this interface. |
| **system-description** | Enables transmission of this device's system description on this interface. |
| **system-name** | Enables transmission of this device's system name on this interface. |
| **and-receive** | Configures this interface to both transmit and receive LLDP packets. |

## Default Values

Be default, all interfaces are configured to transmit and receive LLDP packets of all types.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Functional Notes

Individual LLDP information can be enabled or disabled using the various forms of the **lldp send** command. For example, use the **lldp send and-receive** command to enable transmit and receive of all LLDP information. Then use the **no lldp send port-description** command to prevent LLDP from transmitting port description information.

## Usage Examples

The following example configures the port channel interface to transmit LLDP packets containing all enabled information types:

(config)**#interface port-channel 1**
(config-p-chan 1)**#lldp send**

The following example configures the port channel interface to transmit and receive LLDP packets containing all information types:

(config)**#interface port-channel 1**
(config-p-chan 1)**#lldp send and-receive**

# qos

Use the **qos** command to set the interface to the trusted state and to set the default Class of Service (CoS) value. To return to defaults, use the **no** version of this command. Variations of this command include:

**qos default-cos** <*value*>
**qos trust cos**

## Syntax Description

| | |
|---|---|
| **default-cos** <*value*> | Sets the default CoS value for untrusted ports and all untagged packets. Range is 0 to 7. |
| **trust cos** | Sets the interface to the trusted state. |

## Default Values

By default, the interface is untrusted with a default CoS of 0.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

Set the interface to **trust cos** if received 802.1P CoS values are considered valid (i.e., no need to reclassify) and do not need to be tagged with the default value. When set to untrusted, the **default-cos** value for the interface is used.

## Usage Examples

The following example sets port channel 1 as a trusted interface and assigns untagged packets a CoS value of 1:

(config)**#interface port-channel 1**
(config-p-chan 1)**#qos trust cos**
(config-p-chan 1)**#qos default-cos 1**

# snmp trap

Use the **snmp trap** command to enable all supported Simple Network Management Protocol (SNMP) traps on the interface. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, all interfaces (except virtual Frame Relay interfaces and sub-interfaces) have SNMP traps enabled.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 3.1 | Command was extended to the SHDSL interface. |
| Release 5.1 | Command was expanded to include Ethernet sub-interfaces and Gigabit Ethernet interfaces. |

## Usage Examples

The following example enables SNMP capability on the port channel interface:

(config)**#interface port-channel 1**
(config-p-chan 1)**#snmp trap**

# snmp trap link-status

Use the **snmp trap link-status** command to control the Simple Network Management Protocol (SNMP) variable ifLinkUpDownTrapEnable (RFC2863) to enable (or disable) the interface to send SNMP traps when there is an interface status change. Use the **no** form of this command to disable this trap.

## Syntax Description

No subcommands.

## Default Values

By default, the ifLinkUpDownTrapEnable OID is enabled for all interfaces except virtual Frame Relay interfaces.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 3.1 | Command was extended to the SHDSL interface. |
| Release 5.1 | Command was expanded to include Gigabit Ethernet, port channel, VLAN, E1, and G.703 interfaces. |
| Release 6.1 | Command was expanded to include VLAN and port channel interfaces. |

## Functional Notes

The **snmp trap link-status** command is used to control the RFC2863 ifLinkUpDownTrapEnable OID (OID number 1.3.6.1.2.1.31.1.1.1.14.0).

## Usage Examples

The following example disables the link-status trap on the port channel interface:

(config)**#interface port-channel 1**
(config-p-chan 1)**#no snmp trap link-status**

# spanning-tree bpdufilter

Use the **spanning-tree bpdufilter** command to enable or disable the BPDU filter on a specific interface. This setting overrides the related global setting (refer to *spanning-tree edgeport default* on page 678). Use the **no** version of the command to return to the default setting. Variations of this command include:

**spanning-tree bpdufilter disable**
**spanning-tree bpdufilter enable**

## Syntax Description

| | |
|---|---|
| **disable** | Disables BPDU filter for this interface. |
| **enable** | Enables BPDU filter for this interface. |

## Default Values

By default, this setting is disabled.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

The bpdufilter blocks any BPDUs from being transmitted and received on an interface.

## Usage Examples

The following example enables the BPDU filter on the port channel interface:

(config)**#interface port-channel 3**
(config-p-chan 3)**#spanning-tree bpdufilter enable**

The BPDU filter can be disabled on port channel 3 by issuing the following commands:

(config)**#interface port-channel 3**
(config-p-chan 3)**#spanning-tree bpdufilter disable**

# spanning-tree bpduguard

Use the **spanning-tree bpduguard** command to enable or disable the BPDU guard on a specific interface. This setting overrides the related global setting (refer to *spanning-tree edgeport default* on page 678). Use the **no** version of the command to return to the default setting. Variations of this command include:

**spanning-tree bpduguard disable**
**spanning-tree bpduguard enable**

## Syntax Description

| | |
|---|---|
| **disable** | Disables BPDU guard for this interface. |
| **enable** | Enables BPDU guard for this interface. |

## Default Values

By default, this setting is disabled.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

The bpduguard blocks any BPDUs from being received on an interface.

## Usage Examples

The following example enables the BPDU guard on the port channel interface:

(config)**#interface port-channel 3**
(config-p-chan 3)**#spanning-tree bpduguard enable**

The BPDU guard can be disabled on port channel 3 by issuing the following commands:

(config)**#interface port-channel 3**
(config-p-chan 3)**#spanning-tree bpduguard disable**

# spanning-tree cost <*value*>

Use the **spanning-tree cost** command to assign a cost to the interface. The cost value is used when computing the spanning-tree root path. Use the **no** version of the command to return to the default setting.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies a cost value of 1 to 200,000,000. |

## Default Values

By default, the cost value is set to 1000/(link speed in Mbps).

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following example sets the interface to a path cost of 1200:

(config)**#interface port-channel 3**
(config-p-chan 3)**#spanning-tree cost 1200**

# spanning-tree edgeport

Use the **spanning-tree edgeport** command to configure the interface to be an edgeport. This command overrides the Global setting (refer to *spanning-tree edgeport default* on page 678). Use the **no** version of the command to return to the default setting.

## Syntax Description

No subcommands.

## Default Values

By default, this setting is disabled.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

Enabling this command configures the interface to go to a forwarding state when the link goes up.

## Usage Examples

The following example configures the interface to be an edgeport:

(config)**#interface port-channel 1**
(config-p-chan 1)**#spanning-tree edgeport**

An individual interface can be configured to not be considered an edgeport. For example:

(config)**#interface port-channel 1**
(config-p-chan 1)**#spanning-tree edgeport disable**

or

(config)**#interface port-channel 1**
(config-p-chan 1)**#no spanning-tree edgeport**

# spanning-tree link-type

Use the **spanning-tree link-type** command to configure the spanning tree protocol link type for each interface. Use the **no** version of the command to return to the default setting. Variations of this command include:

**spanning-tree link-type auto**
**spanning-tree link-type point-to-point**
**spanning-tree link-type shared**

## Syntax Description

| | |
|---|---|
| **auto** | Determines link type by the port's duplex settings. |
| **point-to-point** | Manually sets link type to point-to-point, regardless of duplex settings. |
| **shared** | Manually sets link type to shared, regardless of duplex settings. |

## Default Values

By default, the interface is set to **auto**.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

This command overrides the default link type setting determined by the duplex of the individual port. By default, a port configured for half-duplex is set to **shared** link type, and a port configured for full-duplex is set to **point-to-point** link type. Setting the link type manually overrides the default and forces the port to use the specified link type. Use the **link-type auto** command to restore the convention of determining link type based on duplex settings.

## Usage Examples

The following example forces the link type to **point-to-point**, even if the port is configured to be half-duplex:

(config)**#interface port-channel 1**
(config-p-chan 1)**#spanning-tree link-type point-to-point**

## Technology Review

Rapid transitions are possible in RSTP (rapid spanning-tree protocol) by taking advantage of point-to-point links (a port is connected to exactly one other bridge) and edge-port connections (a port is not connected to any additional bridges). Setting the link type to **point-to-point** allows a half-duplex link to act as if it were a point-to-point link.

# spanning-tree port-priority *<value>*

Use the **spanning-tree port-priority** command to select the priority level of this interface. To return to the default setting, use the **no** version of this command.

## Syntax Description

| | |
|---|---|
| *<value>* | Specifies a priority-level value from 0 to 240 (this value must be in increments of 16). |

## Default Values

By default, the **spanning-tree port-priority** is set to 128.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

The only time that this priority level is used is when two interfaces with a path to the root have equal cost. At that point, the level set in this command will determine which port the spanning-tree will use. Set the priority value lower to increase the chance the interface will be used.

## Usage Examples

The following example sets the interface to a priority of **96**:

(config)**#interface port-channel 4**
(config-p-chan 4)**#spanning-tree port-priority 96**

# storm-control action shutdown

Use the **storm-control action shutdown** command to specify that the unit should shutdown when a broadcast, multicast, or unicast storm occurs. To disable the option, use the **no** version of this command.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled; the interface will only filter traffic.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

Enabling this option shuts down the interface if a multicast, unicast, or broadcast storm occurs.

## Usage Examples

The following example shuts down the port channel interface if a storm is detected:

(config)**#interface port-channel 1**
(config-p-chan 1)**#storm-control action shutdown**

# storm-control

Use the **storm-control** command to configure limits on the rates of broadcast, multicast, and unicast traffic on a port. To disable storm-control, use the **no** version of this command. Variations of this command include:

**storm-control broadcast level** *<rising level>*
**storm-control broadcast level** *<rising level> <falling level>*
**storm-control multicast level** *<rising level>*
**storm-control multicast level** *<rising level> <falling level>*
**storm-control unicast level** *<rising level>*
**storm-control unicast level** *<rising level> <falling level>*

## Syntax Description

| | |
|---|---|
| **broadcast level** | Sets levels for broadcast traffic. |
| **multicast level** | Sets levels for multicast traffic. |
| **unicast level** | Sets levels for unicast traffic. |
| *<rising level>* | Specifies a rising level which determines the percentage of total bandwidth the port accepts before it begins blocking packets. Range is 1 to 100 percent. |
| *<falling level>* | Optional. Specifies a falling level which determines when the storm is considered over, causing the AOS to no longer block packets. This level must be less than the rising level. Range is 1 to 100 percent. |

## Default Values

By default, storm control is disabled.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

This setting configures the rising and falling suppression values. When the selected rising level (which is a percentage of total bandwidth) is reached, the port begins blocking packets of the specified type (i.e., broadcast, multicast, or unicast). The AOS uses the rising level as its falling level if no falling level is specified.

## Usage Examples

The following example sets the rising suppression level to **85** percent for multicast packets:

(config)**#interface port-channel 1**
(config-p-chan 1)**#storm-control multicast level 85**

The following example sets the rising suppression level to **80** percent for broadcast packets, with a falling level of **50** percent:

(config)**#interface port-channel 1**
(config-p-chan 1)**#storm-control broadcast level 80 50**

# switchport access vlan *<vlan id>*

Use the **switchport access vlan** command to set the port to be a member of the VLAN when in access mode. To reset the port to be a member of the default VLAN, use the **no** version of this command.

## Syntax Description

| | |
|---|---|
| *<vlan id>* | Specifies a valid VLAN interface ID. Range is 1 to 4094. |

## Default Values

By default, this is set to VLAN 1 (the default VLAN).

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

If the port is in the trunk mode, this command will not alter the switchport mode to access. Instead it will save the value to be applied when the port does switch to access mode. Refer to *switchport mode* on page 1374 for more information.

## Usage Examples

The following example sets the switchport mode to static-access and makes the port channel 1 a member of VLAN 2:

(config)**#interface port-channel 1**
(config-p-chan 1)**#switchport mode access**
(config-p-chan 1)**#switchport access vlan 2**

# switchport gvrp

Use the **switchport gvrp** command to enable GVRP on an interface. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, GVRP is disabled on all ports.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Functional Notes

Enabling GVRP on any interface enables GVRP globally.

## Usage Examples

The following example enables GVRP on port channel 3:

(config)**#interface port-channel 3**
(config-p-chan 3)**#switchport gvrp**

# switchport mode

Use the **switchport mode** command to configure the VLAN membership mode. To reset membership mode to the default value, use the **no** version of this command. Variations of this command include:

**switchport mode access**
**switchport mode trunk**

## Syntax Description

| | |
|---|---|
| **access** | Sets port to be a single (non-trunked) port that transmits and receives no tagged packets. |
| **trunk** | Sets port to transmit and receive packets on all VLANs included within its VLAN allowed list. |

## Default Values

By default, this is set to **access**.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following example sets the port to be a trunk port:

(config)**#interface port-channel 1**
(config-p-chan 1)**#switchport mode trunk**

# switchport trunk allowed vlan

Use the **switchport trunk allowed vlan** command to allow certain VLANs to transmit and receive traffic on this port when the interface is in trunking mode. To return to defaults, use the **no** version of this command. Variations of this command include:

**switchport trunk allowed vlan** <*list*>
**switchport trunk allowed vlan add** <*list*>
**switchport trunk allowed vlan all**
**switchport trunk allowed vlan except** <*list*>
**switchport trunk allowed vlan remove** <*list*>

## Syntax Description

| | |
|---|---|
| **add** | Adds the specified VLAN IDs to the VLAN trunking allowed list. |
| **all** | Adds all configured VLAN IDs to the VLAN trunking allowed list. |
| **except** | Adds all configured VLAN IDs to the VLAN trunking allowed list except those specified in the VLAN ID list. |
| **remove** | Removes VLAN IDs from the VLAN trunking allowed list. |
| <*list*> | Specifies a list of valid VLAN interface IDs. Refer to *Functional Notes* below for additional syntax considerations. |

## Default Values

By default, all valid VLANs are allowed.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

A VLAN list is a set of VLAN IDs. A valid VLAN ID value must be from 1 to 4094 (inclusive). Each VLAN ID in a list is delimited by commas, yet a range of IDs may be expressed as a single element by using a hyphen between endpoints. For example the VLAN ID range **1,2,3,4,6,7,8,9,500** may be more easily expressed as **1-4,6-9,500**. No spaces are allowed in a valid ID range.

## Usage Examples

The following example adds VLANs to the previously existing list of VLANs allowed to transmit and receive on this port:

(config)**#interface port-channel 1**
(config-p-chan 1)**#switchport trunk allowed vlan add 1-4,7-9,500**

# switchport trunk fixed vlan

Use the **switchport trunk fixed vlan** command to change the configured list of VLANs that remain fixed in use only when GVRP is enabled on the interface. Of these VLANs, VLANs statically or dynamically created will be available for use on the interface. Use the **no** form of this command to return to the default setting. Variations of this command include:

**switchport trunk fixed vlan** <*list*>
**switchport trunk fixed vlan add** <*list*>
**switchport trunk fixed vlan all**
**switchport trunk fixed vlan except** <*list*>
**switchport trunk fixed vlan remove** <*list*>

## Syntax Description

| | |
|---|---|
| **add** | Adds VLANs to the VLAN GVRP trunking fixed list. |
| **all** | Adds all VLANs to the VLAN GVRP trunking fixed list. |
| **except** | Adds all VLAN IDs to the VLAN trunking fixed list except those in the command line VLAN ID list. |
| **none** | Removes all VLANs from the VLAN GVRP trunking fixed list. |
| **remove** | Removes VLAN from the VLAN trunking fixed list. |
| <*list*> | Specifies a list of valid VLAN interface IDs. Refer to *Functional Notes* below for additional syntax considerations. |

## Default Values

By default, no VLANs are in the VLAN GVRP trunking fixed list (**switchport trunk fixed vlan none**).

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Functional Notes

This command has no effect on VLAN membership configuration unless GVRP is enabled on the interface. Refer to *gvrp* on page 487 for information on enabling GVRP.

A VLAN list is a set of VLAN IDs. A valid VLAN ID value must be from 1 to 4094 (inclusive). Each VLAN ID in a list is delimited by commas, yet a range of IDs may be expressed as a single element by using a hyphen between endpoints. For example the VLAN ID range **1,2,3,4,6,7,8,9,500** may be more easily expressed as **1-4,6-9,500**. No spaces are allowed in a valid ID range.

## Usage Examples

The following example changes the configured list of fixed VLANs by adding VLAN 50 to the list:

(config-p-chan 1)**#switchport trunk fixed vlan add 1-15,25-30,40**
(config-p-chan 1)#
(config-p-chan 1)**#switchport trunk fixed vlan add 50**
(config-p-chan 1)#

# switchport trunk native vlan <*vlan id*>

Use the **switchport trunk native vlan** command to set the VLAN native to the interface when the interface is in trunking mode. To return to defaults, use the **no** version of this command.

## Syntax Description

| | |
|---|---|
| <*vlan id*> | Specifies a valid VLAN interface ID. Range is 1 to 4094. |

## Default Values

By default, this is set to VLAN 1.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

Configure which VLAN the interface uses as its native VLAN during trunking. Packets from this VLAN leaving the interface will not be tagged with the VLAN number. Any untagged packets received by the interface are considered a part of the native VLAN ID.

## Usage Examples

The following example sets the native VLAN on port channel 1 to VLAN 2:

(config)**#interface port-channel 1**
(config-p-chan 1)**#switchport trunk native vlan 2**

# PPP INTERFACE CONFIGURATION COMMAND SET

To activate the PPP Interface Configuration mode, enter the **interface ppp** command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#interface ppp 1
(config-ppp 1)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:



All other commands for this command set are described in this section in alphabetical order.



# access-policy *<name>*

Use the **access-policy** command to assign a specified access policy to an interface. Access policies are applied to traffic entering an interface. Use the **no** form of this command to remove an access policy association. For more information on using access policies, refer to *ip policy-class <name>* on page 549.

> NOTE *Configured access policies will only be active if the **ip firewall** command has been entered at the Global Configuration mode prompt to enable the AOS security features. All configuration parameters are valid, but no security data processing will be attempted unless the security features are enabled.*

## Syntax Description

| | |
|---|---|
| *<name>* | Identifies the configured access policy by alphanumeric descriptor (all access policy descriptors are case-sensitive). |

## Default Values

By default, there are no configured access policies associated with an interface.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include Ethernet sub-interfaces. |
| Release 6.1 | Command was expanded to include VLAN interfaces. |

## Functional Notes

To assign an access policy to an interface, enter the interface configuration mode for the desired interface and enter **access-policy** *<policy name>*.

## Usage Examples

The following example associates the access policy **Private** (to allow inbound traffic to the Web server) to the PPP interface 1:

Enable the AOS security features:
(config)**#ip firewall**

Associate the access policy with the PPP interface 1:

(config)**#interface ppp 1**
(config-ppp 1)**#access-policy Private**

# alias link<*"text"*>

Use the **alias link** command to provide the management station with an identifying description for each link (PPP physical). Each configured PPP interface (when referenced using SNMP) contains a link (physical port) and a bundle (group of links). RFC1471 (for Link Connection Protocol) provides an interface table to manage lists of bundles and associated links. Use the **no** form of this command to return to the default setting.

## Syntax Description

| | |
|---|---|
| <*"text"*> | Describes the interface (for SNMP) by alphanumeric character string (must be encased in quotation marks). |

## Default Values

By default, the PPP identification string appears as empty quotes. (" ")

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

The **alias link** string should be used to uniquely identify a PPP link. Enter a string that clearly identifies the link.

## Usage Examples

The following example defines a unique character string for the virtual PPP interface (1):

(config)**#interface ppp 1**
(config-ppp 1)**#alias link "PPP_link_1"**

## Technology Review

Please refer to RFC1990 for a more detailed discussion on PPP links and bundles.

# bandwidth *<value>*

Use the **bandwidth** command to provide the bandwidth value of an interface to the higher-level protocols. This value is used in cost calculations. Use the **no** form of this command to restore the default values.

## Syntax Description

| | |
|---|---|
| *<value>* | Specifies the bandwidth value in kbps. Range is 1 to 4,294,967,295 kbps. |

## Default Values

To view default values, use the **show interfaces** command.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Functional Notes

The **bandwidth** command is an informational value that is communicated to the higher-level protocols to be used in cost calculations. This is a routing parameter only and does not affect the physical interface.

## Usage Examples

The following example sets bandwidth of the PPP interface to 10 Mbps:

(config)**#interface ppp 1**
(config-ppp 1)**#bandwidth 10000**

# bridge-group *<number>*

Use the **bridge-group** command to assign an interface to the specified bridge group. This command is supported on all Ethernet interfaces, PPP virtual interfaces, and Frame Relay virtual sub-interfaces. Use the **no** form of this command to remove an interface.

## Syntax Description

| | |
|---|---|
| *<number>* | Specifies the bridge group (by number) to which to assign this interface. Range is 1 to 255. |

## Default Values

By default, there are no configured bridge groups.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

A bridged network can provide excellent traffic management to reduce collisions and limit the amount of bandwidth wasted with unnecessary transmissions when routing is not necessary. Any two interfaces can be bridged (Ethernet to T1 bridge, Ethernet to Frame Relay sub-interface, etc.).

## Usage Examples

The following example assigns the PPP interface to bridge-group 1:

(config)**#interface ppp 1**
(config-ppp 1)**#bridge-group 1**

# bridge-group <*number*> vlan-transparent

Use the **bridge-group vlan-transparent** command to prevent an interface from removing the VLAN tag. Use the **no** form of this command to allow the interface to remove the VLAN tag from the packet.

*The **bridge-group vlan-transparent** command is not a global command. The command must be applied on all interfaces of the bridge group.*

## Syntax Description

| | |
|---|---|
| <*number*> | Specifies the bridge group number. Valid range is 1 to 255. |

## Default Values

By default, VLAN tags are removed from the data.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example prevents the removal of VLAN tags from the packets on the PPP interface labeled 1:

config)**#interface ppp 1**
(config-ppp 1)**#bridge-group 1 vlan-transparent**

# crypto map *<name>*

Use the **crypto map** command to associate crypto maps with the interface. Use the **no** form of this command to remove a crypto map from an interface.

> NOTE *When you apply a map to an interface, you are applying all crypto maps with the given map name. This allows you to apply multiple crypto maps if you have created maps which share the same name but have different map index numbers.*

> NOTE *For VPN configuration example scripts, refer to the technical support note **Configuring VPN** located on the ADTRAN OS System Documentation CD provided with your unit.*

## Syntax Description

| | |
|---|---|
| *<name>* | Assigns a crypto map name to the interface. |

## Default Values

By default, no crypto maps are assigned to an interface.

## Command History

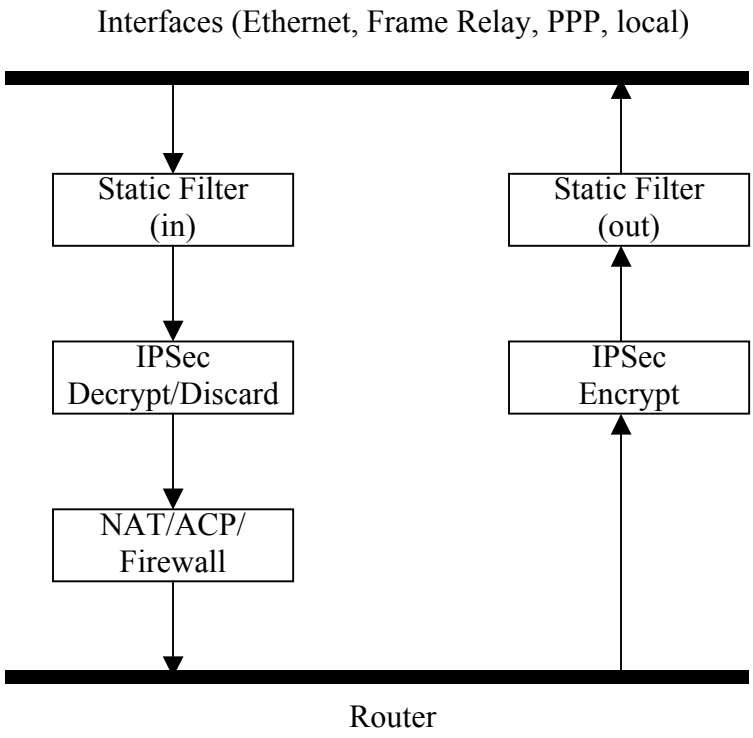
| | |
|---|---|
| Release 4.1 | Command was introduced. |

## Functional Notes

When configuring a system to use both the stateful inspection firewall and IKE negotiation for VPN, keep the following notes in mind.

When defining the policy class and associated access-control lists (ACLs) that describe the behavior of the firewall, do not forget to include the traffic coming into the system over a VPN tunnel terminated by the system. The firewall should be set up with respect to the unencrypted traffic that is destined to be sent or received over the VPN tunnel. The following diagram represents typical AOS data-flow logic.




As shown in the diagram above, data coming into the product is first processed by the static filter associated with the interface on which the data is received. This access group is a true static filter and is available for use regardless of whether the firewall is enabled or disabled. Next (if the data is encrypted) it is sent to the IPSec engine for decryption. The decrypted data is then processed by the stateful inspection firewall. Therefore, given a terminating VPN tunnel, only unencrypted data is processed by the firewall.

The ACLs for a crypto map on an interface work in reverse logic to the ACLs for a policy class on an interface. When specifying the ACLs for a crypto map, the source information is the private local-side, unencrypted source of the data. The destination information will be the far end, unencrypted destination of the data. However, ACLs for a policy class work in reverse. The source information for the ACL in a policy class is the far end. The destination information is the local side.

## Usage Examples

The following example applies all crypto maps with the name **MyMap** to the PPP interface:

(config)**#interface ppp 1**
(config-ppp 1)**#crypto map MyMap**

# dial-backup auto-backup

Use the **dial-backup auto-backup** command to configure the interface to automatically attempt a dial-backup upon failure. Use the **no** form of this command to disable this feature. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of *dial-backup call-mode* on page 1391.

## Syntax Description

No subcommands.

## Default Values

By default, all backup endpoints will automatically attempt dial-backup upon a failure.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include the PPP interface. |

## Usage Examples

The following example enables automatic dial-backup on the endpoint:

(config)**#interface ppp 1**
(config-ppp 1)**#dial-backup auto-backup**

# dial-backup auto-restore

Use the **dial-backup auto-restore** command to configure the interface to automatically discontinue dial-backup when all network conditions are operational. Use the **no** form of this command to disable the auto-restore feature. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of *dial-backup call-mode* on page 1391.

## Syntax Description

No subcommands.

## Default Values

By default, all backup endpoints will automatically restore the primary connection when the failure condition clears.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include the PPP interface. |

## Usage Examples

The following example configures the AOS to automatically restore the primary connection when the failure condition clears:

(config)**#interface ppp 1**
(config-ppp 1)**#dial-backup auto-restore**

# dial-backup backup-delay *<value>*

Use the **dial-backup backup-delay** command to configure the amount of time the router will wait after the failure condition is recognized before attempting to backup the link. Use the **no** form of this command to return to the default value. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of *dial-backup call-mode* on page 1391.

## Syntax Description

| | |
|---|---|
| *<value>* | Specifies the delay period (in seconds) a failure must be active before the AOS will enter backup operation on the interface. Range is 10 to 86,400 seconds. |

## Default Values

By default, the **dial-backup backup-delay** period is set to 10 seconds.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include the PPP interface. |

## Usage Examples

The following example configures the AOS to wait **60** seconds (on an endpoint with an active alarm condition) before attempting dial-backup operation:

(config)**#interface ppp 1**
(config-ppp 1)**#dial-backup backup-delay 60**

# dial-backup call-mode

Use the **dial-backup call-mode** command to specify whether the configured backup interface answers or originates (or a combination of both ) backup calls. Use the **no** form of this command to return to the default value. Variations of this command include:

**dial-backup call-mode answer**
**dial-backup call-mode answer-always**
**dial-backup call-mode originate**
**dial-backup call-mode originate-answer**
**dial-backup call-mode originate-answer-always**

## Syntax Description

| | |
|---|---|
| **answer** | Answers and backs up primary link on failure. |
| **answer-always** | Answers and backs up regardless of primary link state. |
| **originate** | Originates backup call on primary link failure. |
| **originate-answer** | Originates or answers call on primary link failure. |
| **originate-answer-always** | Originates on failure; answers and backs up always. |

## Default Values

By default, the **dial-backup call-mode** is set to **originate-answer**.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include the PPP interface. |

## Functional Notes

The majority of the configuration for AOS dial-backup implementation is configured via the dial-backup PPP interface configuration commands. However, the numbers dialed are configured in the primary interface. Full sample configurations follow:

**Sample configuration for remote router (dialing out)**

```
hostname “Remote3200”
enable password adtran
!
interface eth 0/1
 ip address 192.168.1.254 255.255.255.0
 no shutdown
!
interface modem 1/3
 no shutdown
!
interface t1 1/1
```

```
 coding b8zs
 framing esf
 clock source line
 tdm-group 1 timeslots 1-24
 no shutdown
!
interface fr 1 point-to-point
 frame-relay lmi-type ansi
 no shutdown
 cross-connect 1 t1 1/1 1 fr 1
!
interface fr 1.16 point-to-point
 frame-relay interface-dlci 16
 ip address 10.1.1.2 255.255.255.252
 dial-backup call-mode originate
 dial-backup number 5551111 analog ppp1
 dial-backup number 5552222 analog ppp1
  no shutdown
!
interface ppp 1
 ip address 172.22.56.1 255.255.255.252
 ppp authentication chap
 username remoter outer password remotepass
 ppp chap hostname localrouter
 ppp chap password adtran
 no shutdown
!
ip route 192.168.2.0 255.255.255.0 172.22.56.2 255.255.255.252
!
line telnet 0 4
 password password
```

**Sample configuration for central router (dialing in)**

```
hostname “Central3200”
enable password adtran
!
interface eth 0/1
 ip address 192.168.100.254 255.255.255.0
 no shutdown
!
interface modem 1/3
 no shutdown
!
interface t1 1/1
```

```
 coding b8zs
 framing esf
 clock source line
 tdm-group 1 timeslots 1-24
 no shutdown
!
interface fr 1 point-to-point
 frame-relay lmi-type ansi
 no shutdown
 cross-connect 1 t1 1/1 1 fr 1
!
interface fr 1.100 point-to-point
 frame-relay interface-dlci 100
 ip address 10.1.1.1 255.255.255.252
 dial-backup call-mode answer
 dial-backup number 555-8888 analog ppp 1
!
interface ppp 1
 ip address 172.22.56.2 255.255.255.252
 ppp authentication chap
 username localrouter password adtran
 ppp chap hostname remoterouter
 ppp chap password remotepass
 no shutdown
!
ip route 192.168.1.0 255.255.255.0 172.22.56.1 255.255.255.252

line telnet 0 4
 password password
```

## Usage Examples

The following example configures AOS to generate backup calls for this endpoint using an analog modem interface (to phone number 555 1111) but never answer calls and specifies **ppp 2** as the backup interface:

```
(config)#interface ppp 1
(config-ppp 1)#backup call-mode originate
(config-ppp 1)#backup number 555 1111 analog ppp 2
```

## Technology Review

This technology review provides information regarding specific dial-backup router behavior (i.e., when the router will perform dial-backup, where in the configuration the AOS accesses specific routing information, etc.):

### Dialing Out

1. The AOS determines to place an outbound call when either the Layer 1 or Layer 2 has a failure.
2. When placing outbound calls, the AOS matches the number dialed to a PPP interface. This is accomplished with an addition to the **dial-backup number** command (refer to *dial-backup number <number>* on page 1398).
3. When placing the call, the AOS uses the configuration of the related PPP interface for authentication and IP negotiation.
4. If the call fails to connect on the first number dialed, the AOS places a call to the second number (if a second number is configured). The second number to be dialed references a separate PPP interface.

### Dialing In

1. The AOS receives an inbound call on a physical interface.
2. Caller ID is used to match the **dial-backup number** command to the configured PPP interface.
3. If a match is found, the call connects and the AOS pulls down the primary connection if it is not already in a down state.
4. If no match is found from caller ID, the call is terminated.

# dial-backup connect-timeout <*value*>

Use the **dial-backup connect-timeout** command to specify the number of seconds to wait for a connection after a call is attempted before trying to call again or dialing a different number. It is recommended this number be greater than 60. Use the **no** form of this command to return to the default setting. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of *dial-backup call-mode* on page 1391.

## Syntax Description

| | |
|---|---|
| <*value*> | Selects the amount of time (in seconds) that the router will wait for a connection before attempting another call. Valid range is 10 to 300 seconds. |

## Default Values

By default, the **dial-backup connect-timeout** period is set to 60 seconds.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include the PPP interface. |

## Usage Examples

The following example configures the AOS to wait **120** seconds before retrying a failed dial-backup call:

(config)**#interface ppp 1**
(config-ppp 1)**#dial-backup connect-timeout 120**

# dial-backup force

Use the **dial-backup force** command to manually override the automatic dial-backup feature. This can be used to force a link into backup to allow maintenance to be performed on the primary link without disrupting data. Use the **no** form of this command to return to the normal dial-backup operation state. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of the command *dial-backup call-mode* on page 1391. Variations of this command include:

**dial-backup force backup**
**dial-backup force primary**

## Syntax Description

| | |
|---|---|
| **backup** | Forces backup regardless of primary link state. |
| **primary** | Forces primary link regardless of its state. |

## Default Values

By default, this feature is disabled.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include the PPP interface. |

## Usage Examples

The following example configures the AOS to force this interface into dial-backup:

```
(config)#interface ppp 1
(config-ppp 1)#dial-backup force backup
```

# dial-backup maximum-retry <*value*>

Use the **dial-backup maximum-retry** command to select the number of calls the router will make when attempting to backup a link. Use the **no** form of this command to return to the default state. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of the command *dial-backup call-mode* on page 1391.

## Syntax Description

| | |
|---|---|
| <*value*> | Selects the number of call retries that will be made after a link failure. Valid range is 0 to 15 attempts.<br>Setting this value to **0** will allow unlimited retries during the time the network is failed. |

## Default Values

By default, **dial-backup maximum-retry** is set to 0 attempts.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include the PPP interface. |

## Usage Examples

The following example configures the AOS to retry a dial-backup call four times before considering backup operation not available:

(config)**#interface ppp 1**
(config-ppp 1)**#dial-backup maximum-retry 4**

# dial-backup number <*number*>

Use the **dial-backup number** command to configure the phone number and the call type the router will dial upon network failure. Multiple entries can be made for an interface to allow alternate sites to be dialed. Use the **no** form of this command to return to the default setting. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of the command *dial-backup call-mode* on page 1391. Variations of this command include:

**dial-backup number** <*number*> **analog ppp** <*interface*>
**dial-backup number** <*number*> **digital-56k** <*isdn min chan*> <*isdn max chan*> **ppp** <*interface*>
**dial-backup number** <*number*> **digital 64k** <*isdn min chan*> <*isdn max chan*> **ppp** <*interface*>

## Syntax Description

| | |
|---|---|
| <*number*> | Specifies the phone numbers to call when the backup is initiated. |
| **analog** | Indicates the number connects to an analog modem. |
| **digital-56k** | Indicates the number belongs to a digital 56 kbps per DS0 connection. |
| **digital-64k** | Indicates the number belongs to a digital 64 kbps per DS0 connection. |
| <*isdn min chan*> | Specifies the minimum number of DS0s required for a digital 56 or 64 kbps connection. Range is 1 to 24 DS0s. |
| <*isdn max chan*> | Specifies the maximum number of DS0s desired for a digital 56 or 64 kbps connection. Range is 1 to 24 DS0s. |
| **ppp** <*interface*> | Specifies the PPP interface to use as the backup for this interface. For example, **ppp 1**. |

## Default Values

By default, there are no configured dial-backup numbers.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include the PPP interface. |

## Usage Examples

The following example configures the AOS to dial 704-555-1212 (digital 64 kbps connection) to initiate dial-backup operation for this endpoint using interface PPP 3 backup interface:

(config)**#interface ppp 1**
(config-ppp 1)**#dial-backup number 7045551212 digital-64k 1 1 ppp 3**

# dial-backup priority <*value*>

Use the **dial-backup priority** command to select the backup priority for this interface. This allows the user to establish the highest priority backup link and ensure that link will override backups attempted by lower priority links. Use the **no** form of this command to return to the default value. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of the command *dial-backup call-mode* on page 1391.

## Syntax Description

| | |
|---|---|
| <*value*> | Sets the relative priority of this link. Valid range is 0 to 100. A value of 100 designates the highest priority. |

## Default Values

By default, **dial-backup priority** is set to 50.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include the PPP interface. |

## Usage Examples

The following example assigns the highest priority to this endpoint:

(config)**#interface ppp 1**
(config-ppp 1)**#dial-backup priority 100**

# dial-backup randomize-timers

Use the **dial-backup randomize-timers** command to randomize the call timers to minimize potential contention for resources. Use the **no** form of this command to return to the default value. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of the command *dial-backup call-mode* on page 1391.

## Syntax Description

No subcommands.

## Default Values

By default, the AOS does not randomize the dial-backup call timers.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include the PPP interface. |

## Usage Examples

The following example configures the AOS to randomize the dial-backup timers associated with this endpoint:

(config)**#interface ppp 1**
(config-ppp 1)**#dial-backup randomize-timers**

# dial-backup redial-delay *<value>*

Use the **dial-backup redial-delay** command to configure the delay after an unsuccessful call until the call will be re-tried. Use the **no** form of this command to return to the default setting. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of the command *dial-backup call-mode* on page 1391.

## Syntax Description

| | |
|---|---|
| *<value>* | Specifies the delay in seconds between attempting to re-dial a failed backup attempt. Range is 10 to 3600 seconds. |

## Default Values

By default, **dial-backup redial-delay** is set to 10 seconds.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include the PPP interface. |

## Usage Examples

The following example configures a redial delay of **25** seconds on this endpoint:

(config)**#interface ppp 1**
(config-ppp 1)**#dial-backup redial-delay 25**

# dial-backup restore-delay *<value>*

Use the **dial-backup restore-delay** command to configure the amount of time the router will wait after the network is restored before disconnecting the backup link and reverting to the primary. This setting is used to prevent disconnecting the backup link if the primary link is "bouncing" in and out of alarm. Use the **no** form of this command to return to the default setting. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of the command *dial-backup call-mode* on page 1391.

## Syntax Description

| | |
|---|---|
| *<value>* | Specifies the number of seconds the AOS will wait (after a primary link is restored) before disconnecting dial-backup operation. Range is 10 to 86,400 seconds. |

## Default Values

By default, **dial-backup restore-delay** is set to 10 seconds.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include the PPP interface. |

## Usage Examples

The following example configures the AOS to wait **30** seconds before disconnecting dial-backup operation and restoring the primary connection for this endpoint:

(config)**#interface ppp 1**
(config-ppp 1)**#dial-backup restore-delay 30**

# dial-backup schedule

Use the **dial-backup schedule** command to set the time of day that backup will be enabled. Use this command if backup is desired only during normal business hours and on specific days of the week. Use the **no** form of this command to disable dial-backup (as specified). For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of the command *dial-backup call-mode* on page 1391. Variations of this command include:

**dial-backup schedule day** <*name*>
**dial-backup schedule disable-time** <*value*>
**dial-backup schedule enable-time** <*value*>

## Syntax Description

| | |
|---|---|
| **day** <*name*> | Sets the days to allow backup. Valid range is Monday through Sunday. |
| **disable-time** <*value*> | Sets the time of day to disable backup. Time is entered in 24-hour format (00:00). |
| **enable-time** <*value*> | Sets the time of day to enable backup. Time is entered in 24-hour format (00:00). |

## Default Values

By default, dial-backup is enabled for all days and times if the dial-backup auto-backup command has been issued and the dial-backup schedule has not been entered.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include the PPP interface. |

## Usage Examples

The following example enables dial-backup Monday through Friday 8:00 am to 7:00 pm:

(config)**#interface ppp 1**
(config-ppp 1)**#dial-backup schedule enable-time 08:00**
(config-ppp 1)**#dial-backup schedule disable-time 19:00**
(config-ppp 1)**#no dial-backup schedule day Saturday**
(config-ppp 1)**#no dial-backup schedule day Sunday**

# dynamic-dns

Use the **dynamic-dns** command to configure Dynamic DNS service provided by Dynamic Network Services, Inc. (www.dyndns.org). Use the **no** versions of these commands to disable these features. Variations of this command include:

**dynamic-dns custom** <*hostname*> <*minutes*>
**dynamic-dns dyndns** <*hostname*> <*username*> <*password*>
**dynamic-dns dyndns-custom** <*hostname*> <*username*> <*password*>
**dynamic-dns dyndns-static** <*hostname*> <*username*> <*password*>

## Syntax Description

| | |
|---|---|
| <*hostname*> | Specifies the host name for the server that updates the Dynamic Domain Name Server (DNS). |
| <*minutes*> | Specifies the intervals in minutes to update the server with information (updates also occur when the interface's IP address changes regardless of the update intervals).<br>Refer to *Functional Notes* below for additional argument descriptions. |

## Default Values

No default is necessary for this command.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |
| Release 12.1 | Command was expanded. |

## Functional Notes

**custom** - Constanttime.com's Custom Dynamic DNS$^{SM}$ service allows you complete access and management control over your domain name regardless of where you purchased/registered it. This allows to manage IP address mappings (A records), domain aliases (CNAME records) and mail servers (MX records).

**dyndns** - The Dynamic DNS$^{SM}$ offered by Dynamic Network Services Inc., (DynDNS.org) allows you to alias a dynamic IP address to a static host name in various domains. This allows your unit to be more easily accessed from various locations on the Internet. This service is provided for up to five host names.

**dyndns-custom -** DynDNS.org's Custom DNS$^{SM}$ service provides a full DNS solution, giving you complete control over an entire domain name. A web-based interface provides two levels of control over your domain, catering to average or advanced users. Five globally redundant DNS servers ensure that your domain will always resolve.

A choice of two interfaces is available. The basic interface is designed for most users. It comes preconfigured for the most common configuration and allows for easy creation of most common record types. The advanced interface is designed for system administrators with a solid DNS background, and provides layout and functionality similar to a BIND zone file allowing for the creation of nearly any record type.

Custom DNS$^{SM}$ can be used with both static and dynamic IPs and has the same automatic update capability through Custom DNS-aware clients as Dynamic DNS.

**dyndns-static -** The Static DNS service is similar to DynDNS.org's Dynamic DNS$^{SM}$ service in that it allows a host name such as yourname.dyndns.org to point to your IP address. Unlike a Dynamic DNS host, a Static DNS host does not expire after 35 days without updates, but updates take longer to propagate though the DNS system. This service is provided for up to five host names.

If your IP address does not change often or at all but you still want an easy name to remember it by (without having to purchase your own domain name), Static DNS service is ideal for you.

If you would like to use your own domain name (such as yourname.com), you need Custom DNS service which also provides full dynamic and static IP address support.

**Usage Examples**

The following example sets the dynamic-dns to **dyndns-custom** with host name **host**, user name **user**, and password **pass**:

(config)**#interface ppp 1**
(config-ppp 1)**#dynamic-dns dyndns-custom host user pass**

# fair-queue

Use the **fair-queue** command to enable weighted fair queuing (WFQ) on an interface. Use the **no** form of this command to disable WFQ and enable FIFO queueing for an interface. WFQ is enabled by default for WAN interfaces. Variations of this command include:

**fair-queue**
**fair-queue** *<threshold>*

## Syntax Description

| | |
|---|---|
| *<threshold>* | Optional. Specifies the maximum number of packets that can be present in each conversation sub-queue. Packets received for a conversation after this limit is reached are discarded. Range is 16 to 512 packets. |

## Default Values

By default, **fair-queue** is enabled with a threshold of 64 packets.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following example enables WFQ on the interface with a threshold set at 100 packets:

(config)**#interface ppp 1**
(config-ppp 1)**#fair-queue 100**

# hold-queue <*value*> out

Use the **hold-queue out** command to change the overall size of an interface's WAN output queue. Use the **no** form of this command to return to the default settings.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the total number of packets the output queue can contain before packets are dropped. Range is 16 to 1000 packets. |

## Default Values

The default queue size for WFQ is 400. The default queue size for PPP FIFO and Frame Relay round-robin is 200.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following example sets the overall output queue size to 700:

(config)**#interface ppp 1**
(config-ppp 1)**#hold-queue 700 out**

# ip access-group *<name>*

Use the **ip access-group** command to create an access control list to be used for packets transmitted on or received from the specified interface. Use the **no** form of this command to disable this type of control. Variations of this command include:

**ip access-group** <*name*> **in**
**ip access-group** <*name*> **out**

## Syntax Description

| | |
|---|---|
| *<name>* | Indicates the assigned IP access control list name. |
| **in** | Enables access control on packets received on the specified interface. |
| **out** | Enables access control on packets transmitted on the specified interface. |

## Default Values

By default, these commands are disabled.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Functional Notes

When this command is enabled, the IP destination address of each packet must be validated before being passed through. If the packet is not acceptable per these settings, it is dropped.

## Usage Examples

The following example sets up the router to only allow Telnet traffic into the PPP interface:

(config)**#ip access-list extended TelnetOnly**
(config-ext-nacl)**#permit tcp any any eq telnet**
(config-ext-nacl)**#interface ppp 1**
(config-ppp 1)**#ip access-group TelnetOnly in**

# ip address dhcp

Use the **ip address dhcp** command to use Dynamic Host Configuration Protocol (DHCP) to obtain an address on the interface. Use the **no** form of this command to remove a configured IP address (using DHCP) and disable DHCP operation on the interface. Variations of this command include:

**ip address dhcp**
**ip address dhcp** *<administrative distance>*
**ip address dhcp track** *<name>*
**ip address dhcp track** *<name>* **[***<administrative distance>***]**

## Syntax Description

| | |
|---|---|
| *<administrative distance>* | Optional. Specifies the administrative distance to use when adding the DHCP gateway into the route table. It is used to determine the best route when multiple routes to the same destination exist. The smaller the administrative distance the more reliable the route. Range is 1 to 255. |
| **track** *<name>* | Optional. Attaches a network monitoring track to the DHCP client. The DHCP gateway route for this client will only reside in the route table while the track is in the pass state. For more information on configuring track objects, refer to *track <name>* on page 691. |

## Default Values

By default, the administrative distance value is 1.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |
| Release 8.1 | Command expanded to include PPP interface. |
| Release 13.1 | Command was expanded to include track and administrative distance. |

## Usage Examples

The following example enables DHCP operation on the PPP interface 1:

(config)**#interface ppp 1**
(config-ppp 1)**#ip address dhcp**

The following example enables DHCP operation on the PPP interface 1 and sets the administrative distance as **5**:

(config)**#interface ppp 1**
(config-ppp 1)**#ip address dhcp 5**

# ip address negotiated

Use the **ip address negotiated** command to allow the interface to negotiate (i.e., be assigned) an IP address from the far end PPP connection. Use the **no** form of this command to disable the negotiation for an IP address. Variations of this command include:

**ip address negotiated**
**ip address negotiated** <*administrative distance*>
**ip address negotiated** <*ip address*>
**ip address negotiated** <*ip address*> **no-default**
**ip address negotiated track** <*name*>
**ip address negotiated track** <*name*> **[**<*administrative distance*>**]**

## Syntax Description

| | |
|---|---|
| <*administrative distance*> | Optional. Specifies the administrative distance to use when adding the PPP route to the route table. It is used to determine the best route when multiple routes to the same destination exist. The smaller the administrative distance the more reliable the route. Range is 1 to 255. |
| <*ip address*> | Optional. Specifies a valid IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| **no-default** | Optional. Prevents the insertion of a default route. Some systems already have a default route configured and need a static route to the PPP interface to function correctly. |
| **track** <*name*> | Optional. Attaches a network monitoring track to the PPP interface. The negotiated default route for this client will only reside in the route table while the track is in the pass state. For more information on configuring track objects, refer to *track <name>* on page 691. |

## Default Values

By default, the interface is not assigned an address.

Also by default, the administrative distance value is 1.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |
| Release 13.1 | Command was expanded to include track and administrative distance. |

## Usage Examples

The following example enables the PPP interface to negotiate an IP address from the far end connection:

(config)**#interface ppp 1**
(config-ppp 1)**#ip address negotiated**

The following example enables the PPP interface to negotiate an IP address from the far end connection without inserting a default route:

(config)**#interface ppp 1**
(config-ppp 1)**#ip address negotiated no-default**

# ip address *<ip address> <subnet mask>*

Use the **ip address** command to define an IP address on the specified interface. Use the optional keyword **secondary** to define a secondary IP address. Use the **no** form of this command to remove a configured IP address. Variations of this command include:

**ip address** *<ip address> <subnet mask>*
**ip address** *<ip address> <subnet mask>* **secondary**

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies a valid IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| *<subnet mask>* | Specifies the subnet mask that corresponds to a range of IP addresses (network) or a specific host. Subnet masks can be expressed in dotted decimal notation (for example, **255.255.255.0**) or as a prefix length (for example, **/24**). |
| **secondary** | Optional. Configures a secondary IP address for the specified interface. |

## Default Values

By default, there are no assigned IP addresses.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

Use secondary IP addresses to allow dual subnets on a single interface (when you need more IP addresses than the primary subnet can provide). When using secondary IP addresses, avoid routing loops by verifying that all devices on the network segment are configured with secondary IP addresses on the secondary subnet.

## Usage Examples

The following example configures a secondary IP address of **192.22.72.101/30**:

(config)**#interface ppp 1**
(config-ppp 1)**#ip address 192.22.72.101 255.255.255.252 secondary**

# ip ffe

Use the **ip ffe** command to enable the FastFlow Engine (FFE) on this interface with the default number of entries. Use the **no** form of these commands to disable the FFE. Variations of this command include:

**ip ffe max-entries** <*entries*>

*Issuing this command will cause all FFE entries on this interface to be cleared.*

## Syntax Description

| | |
|---|---|
| **max-entries** <*entries*> | Specifies the maximum number of entries stored in the flow table. Valid range is from 1 to 8192. |

## Default Values

By default, the FFE is disabled. The default number of **max-entries** is 4096.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Functional Notes

The FFE can be used to help reduce routing overhead and thus reduce overall routing times. Routing times are reduced by the creation of a flow table on the ingress interface. The maximum number of entries that can be stored in the flow table at any one time may be specified by using the **max-entries** subcommand. When features such as firewall, VPN, and static filters are enabled, routing time reduction may not be realized.

## Usage Examples

The following example enables the FFE and sets the maximum number of entries in the flow table to 50:

(config)**#interface ppp 1**
(config-ppp 1)**#ip ffe max-entries 50**

## Technology Review

The FastFlow system goal is to increase IP packet throughput by moving as much of the packet processing into the engine as possible. Packets are classified into flows based upon the IP protocol (TCP, UDP, ICMP, etc.), the source and destination IP addresses, IP TOS, and the protocol-specific information such as the source and destination port numbers. Flows are defined as the unidirectional representation of a conversation between two IP hosts. Each ingress interface keeps its own flow table, a collection of flow entries.

The first packet in a flow that is forwarded through the unit will build a flow entry. When a flow entry is looked up but no entry is found, a FastFlowBuilder object is allocated and attached to the packet. As the packet passes through the various processing layers, each subsystem will add processing to the FastFlowBuilder. When packet is about to be forwarded out of the egress interface, the FastFlowBuilder will be finalized. That is, the flow entry being built will be checked for completeness and committed to the flow table on the ingress interface. Subsequent flow matches can then bypass the normal processing layers.

# ip helper-address *<ip address>*

Use the **ip helper-address** command to configure the AOS to forward User Datagram Protocol (UDP) broadcast packets received on the interface. Use the **no** form of this command to disable forwarding packets.

> NOTE *The **ip helper** command must be used in conjunction with the **ip forward-protocol** command to configure the AOS to forward UDP broadcast packets. Refer to ip forward-protocol udp <value> on page 532 for more information.*

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies the destination IP address for the forwarded UDP packets. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

## Default Values

By default, broadcast UDP packets are not forwarded.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

When used in conjunction with the **ip forward-protocol** command, the **ip helper-address** feature allows you to customize which broadcast packets are forwarded.

To implement the helper address feature, assign a helper address(es) (specifying the device that needs to receive the broadcast traffic) to the interface closest to the host that transmits the broadcast packets. When broadcast packets (of the specified type forwarded using the **ip forward-protocol** command) are received on the interface, they will be forwarded to the device that needs the information.

Only packets meeting the following criteria are considered eligible by the **ip helper-address** feature:

1. The packet IP protocol is User Datagram Protocol (UDP).
2. Any UDP port specified using the **ip forward-protocol** command.
3. The media access control (MAC) address of the frame is an all-ones broadcast address (ffff.ffff.ffff).
4. The destination IP address is broadcast defined by all ones (255.255.255.255) or a subnet broadcast (for example, 192.33.4.251 for the 192.33.4.248/30 subnet).

## Usage Examples

The following example forwards all DNS broadcast traffic to the DNS server with IP address 192.33.5.99:

(config)**#ip forward-protocol udp domain**
(config)**#interface ppp 1**
(config-ppp 1)**#ip helper-address 192.33.5.99**

# ip igmp

Use the **ip igmp** command to configure multicasting-related functions for the interface. Variations of this command include:

**ip igmp immediate-leave**
**ip igmp last-member-query-interval** <*milliseconds*>
**ip igmp querier-timeout** <*seconds*>
**ip igmp query-interval** <*seconds*>
**ip igmp query-max-response-time** <*seconds*>
**ip igmp static-group** <*address*>
**ip igmp version [1 | 2]**

## Syntax Description

| | |
|---|---|
| **immediate-leave** | Specifies that if only one host (or IGMP snooping switch) is connected to the interface, when a leave is received, multicast of that group is immediately terminated as opposed to sending a group query and timing out the group if no device responds. Works in conjunction with **ip igmp last-member-query-interval**. Applies to all groups when configured. Use the **no** form of this command to disable the **immediate-leave** feature. |
| **last-member-query-interval** <*milliseconds*> | Controls the timeout (in milliseconds) used to detect whether any group receivers remain on an interface after a receiver leaves a group. If a receiver sends a leave-group message (IGMP Version 2), the router sends a group-specific query on that interface. After twice the time specified by this command plus as much as one second longer, if no receiver responds, the router removes that interface from the group and stops sending that group's multicast packets to the interface. Range is 100 to 65,535 ms. Use the **no** form of this command to return to the default setting. |
| **querier-timeout** <*seconds*> | Specifies the interval (in seconds) that the router waits after the current querier's last query before it takes over as querier (IGMP V2). Range is 60 to 300 seconds. Use the **no** form of this command to return to the default setting. |
| **query-interval** <*seconds*> | Specifies the interval (in seconds) at which IGMP queries are sent on an interface. Host query messages are addressed to the all-hosts multicast group with an IP TTL of 1. The router uses queries to detect whether multicast group members are on the interface and to select an IGMP designated router for the attached segment (if more than one multicast router exists). Only the designated router for the segment sends queries. For IGMP V2, the designated router is the router with the lowest IP address on the segment. Range is 0 to 65,535 seconds. Use the **no** form of this command to return to the default setting. |
| **query-max-response-time** <*seconds*> | Specifies the maximum response time (in seconds) advertised by this interface in queries when using IGMP V2. Hosts are allowed a random time within this period to respond, reducing response bursts. Use the **no** form of this command to return to the default setting. |

## Syntax Description

| | |
|---|---|
| **static-group** <*address*> | Configures the router's interface to be a statically-connected member of the specified group. Packets received on the correct RPF interface are forwarded to this interface regardless of whether any receivers have joined the specified group using IGMP. Use the **no** form of this command to remove a configured static group. |
| **version [1 \| 2]** | Sets the interface's IGMP version. Use the **no** form of this command to return to the default setting. |

## Default Values

| | |
|---|---|
| **ip igmp immediate-leave** | No default |
| **ip igmp last-member-query-interval** | 1000 milliseconds |
| **ip igmp querier-timeout** | 2x the **query-interval** value |
| **ip igmp query-interval** | 60 seconds |
| **ip igmp query-max-response-time** | 10 seconds |
| **ip igmp static-group** | No default |
| **ip igmp version** | Version 1 |

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |

## Usage Examples

The following example sets the query message interval on the interface to 200 milliseconds:

(config)**#interface ppp 1**
(config-ppp 1)**#ip igmp last-member-query-interval 200**

# ip mcast-stub downstream

Use the **ip mcast-stub downstream** command to enable multicast forwarding and IGMP (router mode) on an interface and place it in multicast stub downstream mode. Use the **no** form of this command to disable.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |

## Functional Notes

This command is used in IP multicast stub applications in conjunction with the **ip mcast-stub helper-address** and **ip mcast-stub upstream** commands. Downstream interfaces connect to segments with multicast hosts. Multiple interfaces may be configured in downstream mode; however, interfaces connecting to the multicast network (upstream) should not be configured in downstream mode. Interfaces configured as downstream should have the lowest IP address of all IGMP-capable routers on the connected segment in order to be selected as the designated router and ensure proper forwarding. Refer to *ip mcast-stub helper-address <ip address>* on page 546 and *ip mcast-stub upstream* on page 1422 for more information.

## Usage Examples

The following example enables multicast forwarding and IGMP on the interface:

```
(config)#interface ppp 1
(config-ppp 1)#ip mcast-stub downstream
```

# ip mcast-stub fixed

Use the **ip mcast-stub fixed** command to allow forwarding of multicast traffic on a selected interface after enabling multicast routing. Use the **no** form of this command to disable this mode.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Functional Notes

Multicast routing must be enabled prior to setting **ip mcast-stub fixed** on the selected interface. Also, use the **ip igmp static-group** <*A.B.C.D*> command to receive multicast traffic without host-initiated Internet Group Management Protocol (IGMP) activity on the selected interface. Otherwise, all host-initiated IGMP transactions will enter multicast routes on the router's interface involved with IGMP activities.

### Usage Examples

The following example enables multicast traffic forwarding and IGMP on the interface:

(config)**#interface ppp 1**
(config-ppp 1)**#ip mcast-stub fixed**

# ip mcast-stub helper-enable

Use the **ip mcast-stub helper-enable** command to assign the **ip mcast-stub helper-address** as the IGMP proxy. Use the **no** form of this command to disable.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |
| Release 10.1 | Command was expanded to include PPP interfaces. |

## Functional Notes

This command is used in IP multicast stub applications in conjunction with the **ip mcast-stub helper-address**, **ip mcast-stub upstream**, and **ip mcast-stub downstream** commands. When enabled, the interface becomes a helper forwarding interface. The IGMP host function is dynamically enabled and the interface becomes the active upstream interface, enabling the unit to perform as an IGMP proxy. Refer to *ip mcast-stub helper-address <ip address>* on page 546, *ip mcast-stub downstream* on page 1419, and *ip mcast-stub upstream* on page 1422 for more information.

## Usage Examples

The following example sets the helper address as the IGMP proxy:

(config)**#interface ppp 1**
(config-ppp 1)**#ip mcast-stub helper-enable**

# ip mcast-stub upstream

Use the **ip mcast-stub upstream** command to enable multicast forwarding on an interface and place it in multicast stub upstream mode. Use the **no** form of this command to disable.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |

## Functional Notes

This command is used in IP multicast stub applications in conjunction with the **ip mcast-stub helper-address** and **ip mcast-stub downstream** commands. When enabled, the interface becomes a candidate to be a helper forwarding interface. If chosen as the best path toward the helper address by the router's unicast route table, the IGMP host function is dynamically enabled and the interface becomes the active upstream interface, enabling the router to perform as an IGMP proxy. Though multiple interfaces may be candidates, no more than one interface will actively serve as the helper forwarding interface. Refer to *ip mcast-stub helper-address <ip address>* on page 546 and *ip mcast-stub downstream* on page 1419 for more information.

## Usage Examples

The following example enables multicast forwarding on the interface:

(config)**#interface ppp 1**
(config-ppp 1)**#ip mcast-stub upstream**

# ip ospf

Use the **ip ospf** command to customize OSPF settings (if needed). Use the **no** form of these commands to return to the default settings. Variations of this command include:

**ip ospf authentication-key** <*password*>
**ip ospf cost** <*value*>
**ip ospf dead-interval** <*seconds*>
**ip ospf hello-interval** <*seconds*>
**ip ospf message-digest-key [1 | 2] md5** <*key*>
**ip ospf priority** <*value*>
**ip ospf retransmit-interval** <*seconds*>
**ip ospf transmit-delay** <*seconds*>

## Syntax Description

| | |
|---|---|
| **authentication-key** <*password*> | Assigns a simple-text authentication password to be used by other routers using the OSPF simple password authentication. |
| **cost** <*value*> | Specifies the OSPF cost of sending a packet on the interface. This value overrides any computed cost value. Range is 1 to 65,535. |
| **dead-interval** <*seconds*> | Sets the maximum interval (in seconds) allowed between hello packets. If the maximum is exceeded, neighboring devices will determine that the device is down. Range is 0 to 32,767 seconds. |
| **hello-interval** <*seconds*> | Specifies the interval (in seconds) between hello packets sent on the interface. Range is 0 to 32,767 seconds. |
| **message-digest-key [1 \| 2] md5** <*key*> | Configures OSPF Message Digest 5 (MD5) authentication (16-byte max) keys. |
| **priority** <*value*> | Sets the OSPF priority. The value set in this field helps determine the designated router for this network. Range is 0 to 255. |
| **retransmit-interval** <*seconds*> | Specifies the interval (in seconds) between link-state advertisements (LSAs). Range is 0 to 32,767 seconds. |
| **transmit-delay** <*seconds*> | Sets the estimated time (in seconds) required to send an LSA on the interface. Range is 0 to 32,767 seconds. |

## Default Values

| | |
|---|---|
| **dead-interval** <*seconds*> | 40 seconds |
| **hello-interval** <*seconds*> | 10 seconds: Ethernet, point-to-point, Frame Relay, and PPP |
| **retransmit-interval** <*seconds*> | 5 seconds |
| **transmit-delay** <*seconds*> | 1 second |

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Usage Examples

The following example sets the maximum number of seconds allowed between hello packets to 25,000:

(config)**#interface ppp 1**
(config-ppp 1)**#ip ospf dead-interval 25000**

# ip ospf authentication

Use the **ip ospf authentication** command to authenticate an interface that is performing OSPF authentication. Use the **no** form of this command to return to the default setting. Variations of this command include:

**ip ospf authentication**
**ip ospf authentication message-digest**
**ip ospf authentication null**

## Syntax Description

| | |
|---|---|
| **message-digest** | Optional. Selects message-digest authentication type. |
| **null** | Optional. Specifies that no authentication be used. |

## Default Values

By default, **ip ospf authentication** is set to null (meaning no authentication is used).

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Usage Examples

The following example specifies that no authentication will be used on the PPP interface:

(config)**#interface ppp 1**
(config-ppp 1)**#ip ospf authentication null**

# ip ospf network

Use the **ip ospf network** command to specify the type of network on this interface. Use the **no** form of this command to return to the default setting. Variations of this command include:

**ip ospf network broadcast**
**ip ospf network point-to-point**

## Syntax Description

| | |
|---|---|
| **broadcast** | Sets the network type for broadcast. |
| **point-to-point** | Sets the network type for point-to-point. |

## Default Values

By default, Ethernet defaults to broadcast. PPP and Frame Relay default to point-to-point.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Functional Notes

A point-to-point network will not elect designated routers.

## Usage Examples

The following example designates a broadcast network type:

(config)**#interface ppp 1**
(config-ppp 1)**#ip ospf network broadcast**

# ip pim sparse-mode

Use the **ip pim sparse-mode** command to enable protocol-independent multicast (PIM) sparse mode for this interface. Use the **no** form of this command to disable PIM sparse mode.

## Syntax Description

No subcommands.

## Default Values

By default, PIM sparse mode for this interface is disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Functional Notes

PIM Sparse Mode is a multicast routing protocol that makes use of the unicast forwarding table. It builds unidirectional shared trees rooted at a Rendezvous Point (RP) for a multicast group or a shortest-path tree rooted at a specific source for a multicast group.

## Usage Examples

The following example enables PIM sparse mode on the interface:

```
(config)#interface ppp 1
(config-ppp 1)#ip pim sparse-mode
```

# ip pim-sparse dr-priority <*value*>

Use the **ip pim-sparse dr-priority** command to specify the priority for the designated router (DR). This command modifies the routers priority in the DR election process. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the priority of this interface (to be used when determining the DR). Valid range is 1 to 4,294,967,295. |

## Default Values

By default, the priority of all PIM interfaces is 1.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Functional Notes

Interfaces advertise their configured priority values in the hello messages transmitted on the interface. Routers use the priority values to determine the appropriate DR. The router on the network segment with the highest priority is selected as the DR. If a hello message is received on the interface from a router on the network segment and it does not contain a priority, the entire network segment defaults to DR selection based on IP addresses instead of priority. In this instance, the DR is selected as the router on the network segment that has the highest IP address. AOS will always include a priority in all transmitted hello messages. If no priority is specifically designated by the user, the priority is set as the default of 1.

## Usage Examples

The following example specifies a priority of **100** on the PPP 1 interface:

(config)**#interface ppp 1**
(config-ppp 1)**#ip pim-sparse dr-priority 100**

# ip pim-sparse hello-timer <*value*>

Use the **ip pim-sparse hello-timer** command to specify protocol-independent multicast (PIM) sparse hello timer period. This is the time interval at which periodic hellos are sent out on all interfaces of a PIM-capable router. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the interval (in seconds) at which periodic hellos are sent out of the interface. Valid range is 10 to 3600 seconds. |

## Default Values

By default, the hellos are transmitted on PIM interfaces every 60 seconds.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Functional Notes

Hello messages are used to inform neighbors of a router's presence. Hello messages normally generate a small amount of traffic on an interface. Setting the **hello-timer** to a small interval increases the number of hellos sent (thus increasing the amount of traffic). Set the **hello-timer** to a reasonable value, taking into consideration the bandwidth available on the interface.

## Usage Examples

The following example specifies hellos be sent on the PPP 1 interface every **3600** seconds:

(config)**#interface ppp 1**
(config-ppp 1)**#ip pim-sparse hello-timer 3600**

# ip pim-sparse nbr-timeout <*value*>

Use the **ip pim-sparse nbr-timeout** command to specify protocol-independent multicast (PIM) sparse neighbor timeout. This is the time interval after which a PIM-capable router will consider a neighbor not present. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the time interval in seconds after which a neighbor is considered not present. Valid range is 30 to 10,800 seconds. |

## Default Values

By default, the nbr-timeout is set to 105 seconds.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example sets the nbr-timeout to 300 seconds:

(config)**#interface ppp 1**
(config-ppp 1)**#ip pim-sparse nbr-timeout 300**

# ip pim-sparse override-interval <*value*>

Use the **ip pim-sparse override-interval** command to specify the protocol-independent multicast (PIM) sparse join/prune override interval. This delay interval is the period after a join/prune that another router on the LAN may override a join/prune. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the delay time in milliseconds. Valid range is 0 to 65,535 milliseconds. |

## Default Values

By default, the override-interval is set to 2500 milliseconds.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example sets the override-interval to 3000 milliseconds:

(config)**#interface ppp 1**
(config-ppp 1)**#ip pim-sparse override-interval 3000**

# ip pim-sparse propagation-delay <*value*>

Use the **ip pim-sparse propagation-delay** command to specify the expected propagation delay for join/prune messages. Set the propagation delay (in milliseconds) to estimate the amount of delay found in the local link. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the expected propagation delay in the local link in milliseconds. Valid range is 0 to 32,767 milliseconds. |

## Default Values

By default, the propagation delay is set to 500 milliseconds.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example sets the propagation delay to **300** milliseconds:

(config)**#interface ppp 1**
(config-ppp 1)**#ip pim-sparse propagation-delay 300**

# ip policy route-map *<name>*

Use the **ip policy route-map** command to assign a policy route-map to this interface. Use the **no** form of this command to remove the route-map policy.

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies the name of the policy route map to assign to this interface. |

## Default Values

By default, no policy route map is assigned to this interface.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example assigns the policy route map **policy1** to the interface:

(config)**#interface ppp 1**
(config-ppp 1)**#ip policy route-map policy1**

# ip proxy-arp

Use the **ip proxy-arp** to enable proxy Address Resolution Protocol (ARP) on the interface. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, proxy ARP is enabled.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

In general, the principle of proxy ARP allows a router to insert its IP address in the source IP address field of a packet (if the packet is from a host on one of its subnetworks). This allows hosts to reach devices on other subnetworks without implementing routing or specifying a default gateway.

If proxy ARP is enabled, the AOS will respond to all proxy ARP requests with its specified MAC address and forward packets accordingly.

Enabling proxy ARP on an interface may introduce unnecessary ARP traffic on the network.

## Usage Examples

The following enables proxy ARP on the virtual PPP interface:

(config)**#interface ppp 1**
(config-ppp 1)**#ip proxy-arp**

# ip rip receive version

Use the **ip rip receive version** command to configure the RIP version the unit accepts in all RIP packets received on the interface. Use the **no** form of this command to restore the default value. Variations of this command include:

**ip rip receive version 1**
**ip rip receive version 2**

## Syntax Description

| | |
|---|---|
| **1** | Accepts only received RIP version 1 packets on the interface. |
| **2** | Accepts only received RIP version 2 packets on the interface. |

## Default Values

By default, all interfaces implement RIP version 1 (the default value for the **version** command).

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

Use the **ip rip receive version** to specify a RIP version that overrides the **version** (in the Router RIP) configuration.

The AOS only accepts one version (either 1 or 2) on a given interface.

## Usage Examples

The following example configures the virtual PPP interface to accept only RIP version 2 packets:

(config)**#interface ppp 1**
(config-ppp 1)**#ip rip receive version 2**

# ip rip send version

Use the **ip rip send version** command to configure the RIP version the unit sends in all RIP packets transmitted on the interface. Use the **no** form of this command to restore the default value. Variations of this command include:

**ip rip send version 1**
**ip rip send version 2**

## Syntax Description

| | |
|---|---|
| **1** | Transmits only RIP version 1 packets on the interface. |
| **2** | Transmits only RIP version 2 packets on the interface. |

## Default Values

By default, all interfaces transmit RIP version 1 (the default value for the **version** command).

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

Use the **ip rip send version** to specify a RIP version that overrides the **version** (in the Router RIP) configuration.

The AOS only transmits one version (either 1 or 2) on a given interface.

## Usage Examples

The following example configures the virtual PPP interface to transmit only RIP version 2 packets:

(config)**#interface ppp 1**
(config-ppp 1)**#ip rip send version 2**

# ip rip summary-address *<ip address> <subnet mask>*

Use the **ip rip summary-address** command to manually summarize the routes Routing Information Protocol (RIP) will advertise and send out a specified interface. Use the **no** form of this command to disable this mode.

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies the summarized network IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.0**). |
| *<network mask>* | Specifies the subnet mask that corresponds to the range of IP addresses (network). Subnet masks can be expressed in dotted decimal notation (for example, **255.255.255.0**) or as a prefix length (for example, **/24**). |

## Default Values

By default, no manual summarization is applied by RIP.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Functional Notes

Unlike the automatic summarization on classful network boundaries, only specific network advertisements are made by RIP using the **ip rip summary-address** command. This command is only effective if RIP version 2 is configured.

## Usage Examples

The following example enables manual summarization on the specified IP address:

(config)**#interface ppp 1**
(config-ppp 1)**#ip rip summary-address 10.10.123.0 255.255.255.0**

# ip route-cache

Use the **ip route-cache** command to enable fast-cache switching on the interface. Use the **no** form of this command to disable fast-cache switching and return to process switching mode.

*Using Network Address Translation (NAT) or the AOS firewall capabilities on an interface requires process switching mode (using the **no ip route-cache** command).*

## Syntax Description

No subcommands.

## Default Values

By default, fast-cache switching is enabled on all Ethernet and virtual Frame Relay sub-interfaces. IP route cache is enabled for all virtual PPP interfaces.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |

## Functional Notes

Fast-cache switching allows an IP interface to provide optimum performance when processing IP traffic.

## Usage Examples

The following example enables fast-cache switching on the virtual PPP interface:

(config)**#interface ppp 1**
(config-ppp 1)**#ip route-cache**

# ip unnumbered *<interface>*

Use the **ip unnumbered** command to use the IP address assigned to the specified interface for all IP processing on the active interface. Use the **no** form of this command to remove the unnumbered configuration.

## Syntax Description

| | |
|---|---|
| *<interface>* | Specifies the interface that contains the IP address to use as the source address for all packets transmitted on this interface. Specify an interface in the format *<interface type [slot/port \| slot/port.sub-interface id \| interface id \| interface id.sub-interface id]>*. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **show ip unnumbered ?** for a list of valid interfaces. |

## Default Values

By default, all interfaces are configured to use a specified IP address (using the **ip address** command).

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 11.1 | Command was expanded to include demand interfaces. |

## Functional Notes

If **ip unnumbered** is enabled on an interface, all IP traffic from the interface will use a source IP address taken from the specified interface. For example, specifying **ip unnumbered eth 0/1** while in the PPP Interface Configuration mode configures the PPP interface to use the IP address assigned to the Ethernet interface for all IP processing. In addition, the AOS uses the specified interface information when sending route updates over the unnumbered interface. Static routes may either use the interface name (ppp 1) or the far-end address (if it will be discovered).

## Usage Examples

The following example configures the PPP interface (labeled **ppp 1**) to use the IP address assigned to the Ethernet interface (**eth 0/1**):

(config)**#interface ppp 1**
(config-ppp 1)**#ip unnumbered eth 0/1**

# ip urlfilter <*name*>

Use the **ip urlfilter** command to apply a URL filter to the interface for all inbound or outbound traffic. Use the **no** form of this command to remove the URL filter from an interface. Variations of this command include:

**ip urlfilter** <*name*> **in**
**ip urlfilter** <*name*> **out**

## Syntax Description

| | |
|---|---|
| <*name*> | Specifies the URL filter name to use on the interface. |
| **in** | Applies the filter to the inbound traffic. |
| **out** | Applies the filter to the outbound traffic. |

## Default Values

By default, there are no URL filters applied to any interfaces.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Functional Notes

The firewall must be enabled using the **ip firewall** command in order to use URL filters. The URL filter must be created by using the **ip urlfilter** <*filtername*> **http** command before applying it to the interface. See *ip urlfilter <name> http* on page 598 for more information on using this command.

## Usage Example

The following example performs URL filtering on all traffic entering through the PPP interface (labeled **ppp 1**) and matches the URL filter named **MyFilter**:

(config)**#interface ppp 1**
(config-ppp 1)**#ip urlfilter MyFilter in**

# keepalive <*value*>

Use the **keepalive** command to enable the transmission of keepalive packets on the interface and specify the time interval in seconds between transmitted packets. Use the **no** form of this command to return to the default setting.

## Syntax Description

| | |
|---|---|
| <*value*> | Defines the time interval (in seconds) between transmitted keepalive packets. Valid range is 0 to 32,767 seconds. |

## Default Values

By default, the time interval between transmitted keepalive packets is 10 seconds.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

If three keepalive packets are sent to an interface with no response, the interface is considered down. To detect interface failures quickly, specify a smaller keepalive time.

## Usage Examples

The following example specifies a keepalive time of 5 seconds on the virtual PPP interface:

(config)**#interface ppp 1**
(config-ppp 1)**#keepalive 5**

# lldp receive

Use the **lldp receive** command to allow LLDP packets to be received on this interface. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, all interfaces are configured to send and receive LLDP packets.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Usage Examples

The following example configures the PPP interface to receive LLDP packets:

```
(config)#interface ppp 1
(config-ppp 1)#lldp receive
```

# lldp send

Use the **lldp send** command to configure this interface to transmit LLDP packets or to control the types of information contained in the LLDP packets transmitted by this interface. Use the **no** form of these commands to disable these features. Variations of this command include:

**lldp send management-address**
**lldp send port-description**
**lldp send system-capabilities**
**lldp send system-description**
**lldp send system-name**
**lldp send-and-receive**

## Syntax Description

| | |
|---|---|
| **management-address** | Enables transmission of management address information on this interface. |
| **port-description** | Enables transmission of port description information on this interface. |
| **system-capabilities** | Enables transmission of this device's system capabilities on this interface. |
| **system-description** | Enables transmission of this device's system description on this interface. |
| **system-name** | Enables transmission of this device's system name on this interface. |
| **and-receive** | Configures this interface to both transmit and receive LLDP packets. |

## Default Values

Be default, all interfaces are configured to transmit and receive LLDP packets of all types.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Functional Notes

Individual LLDP information can be enabled or disabled using the various forms of the **lldp send** command. For example, use the **lldp send and-receive** command to enable transmit and receive of all LLDP information. Then use the **no lldp send port-description** command to prevent LLDP from transmitting port description information.

## Usage Examples

The following example configures the PPP interface to transmit LLDP packets containing all enabled information types:

(config)**#interface ppp 1**
(config-ppp 1)**#lldp send**

The following example configures the PPP interface to transmit and receive LLDP packets containing all information types:

(config)**#interface ppp 1**
(config-ppp 1)**#lldp send and-receive**

# max-reserved-bandwidth <*value*>

Use the **max-reserved-bandwidth** command to specify the percentage of interface bandwidth reserved for use in user-defined (priority or class-based) queues. The remainder of the interface bandwidth is reserved for system critical traffic and is not available to user-defined queues. Use the **no** form of this command to restore the default values.


*Reserving a portion of the interface bandwidth for system critical traffic is necessary for proper operation. Specifying the entire interface bandwidth for use in user-defined queues can cause undesirable operation.*

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the maximum percentage of bandwidth to reserve for QoS. This setting is configured as a percentage of the total interface speed. Range: 1 to 100 percent. |

## Default Values

By default, **max-reserved-bandwidth** is set to 75 percent, which reserves 25 percent of the interface bandwidth for system critical traffic.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example specifies **85** percent of the bandwidth on the PPP 1 be available for use in user-defined queues:

(config)**#interface ppp 1**
(config-ppp 1)**#max-reserved-bandwidth 85**

# media-gateway ip

Use the **media-gateway ip** command to associate an IP address source to use for RTP traffic. When configuring VoIP, RTP traffic needs an IP address to be associated with it. However, some interfaces allow "dynamic" configuration of IP addresses, and thus, this value could change periodically. Use the **no** form of these commands to disable these functions. Variations of this command include:

**media-gateway ip loopback** <*ip address*>
**media-gateway ip primary**
**media-gateway ip secondary** <*ip address*>

## Syntax Description

| | |
|---|---|
| **loopback** <*ip address*> | Use an IP address statically defined to a loopback interface. Helpful when using a single IP address across multiple WAN interfaces for RTP traffic. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| **primary** | Use the IP address that is configured as primary on this interface. Applies to static, DHCP, or negotiated addresses. |
| **secondary** <*ip address*> | Use the statically defined secondary IP address of this interface to be used for RTP traffic. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

## Default Values

By default, **media-gateway ip** is disabled.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example configures the unit to use the primary IP address for RTP traffic:

(config)**#interface ppp 1**
(config-ppp 1)**#media-gateway ip primary**

# mtu *<size>*

Use the **mtu** command to configure the maximum transmit unit (MTU) size for the active interface. Use the **no** form of this command to return to the default value.

## Syntax Description

| | | |
|---|---|---|
| *<size>* | Configures the window size for transmitted packets. The valid ranges for the various interfaces are listed below: | |
| | ATM interfaces | 64 to 1520 |
| | Demand interfaces | 64 to 1520 |
| | Ethernet interfaces | 64 to 1500 |
| | FDL interfaces | 64 to 256 |
| | HDLC interfaces | 64 to 1520 |
| | Loopback interfaces | 64 to 1500 |
| | Tunnel interfaces | 64 to 18,190 |
| | Virtual Frame Relay sub-interfaces | 64 to 1520 |
| | Virtual PPP interfaces | 64 to 1500 |

## Default Values

| | | |
|---|---|---|
| *<size>* | The default values for the various interfaces are listed below: | |
| | ATM interfaces | 1500 |
| | Demand interfaces | 1500 |
| | Ethernet interfaces | 1500 |
| | FDL interfaces | 256 |
| | HDLC interfaces | 1500 |
| | Loopback interfaces | 1500 |
| | Tunnel interfaces | 1500 |
| | Virtual Frame Relay sub-interfaces | 1500 |
| | Virtual PPP interfaces | 1500 |

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

OSPF will not become adjacent on links where the MTU sizes do not match. If router A and router B are exchanging hello packets but their MTU sizes do not match, they will never reach adjacency. This is by design and required by the RFC.

## Usage Examples

The following example specifies an MTU of 1200 on the virtual PPP interface:

(config)**#interface ppp 1**
(config-ppp 1)**#mtu 1200**

# peer default ip address *<ip address>*

Use the **peer default ip address** command to specify the default IP address of the remote end of this interface. Use the **no** form of this command to remove an assigned IP address.

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies the default IP address for the remote end. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

## Default Values

By default, there is no assigned peer default IP address.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Functional Notes

This command is useful if the peer does not send the IP address option during PPP negotiations.

## Usage Examples

The following example sets the default peer IP address to 192.22.71.50:

(config)**#interface ppp 1**
(config-ppp 1)**#peer default ip address 192.22.71.50**

# ppp authentication

Use the **ppp authentication** command to specify the authentication protocol on the PPP virtual interface that the peer should use to authenticate itself. Use the **no** form of this command to disable this feature. Variations of this command include:

**ppp authentication chap**
**ppp authentication pap**

## Syntax Description

| | |
|---|---|
| **chap** | Configures CHAP authentication on the interface. |
| **pap** | Configures PAP authentication on the interface. |

## Default Values

By default, PPP endpoints have no authentication configured.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Technology Review

CHAP and PAP are two authentication methods that enjoy widespread support. Both methods are included in the AOS and are easily configured.

*The authentication method set up on the local router can be different from that on the peer. Also, just because one router requires authentication from its peer does not mean it also has to authenticate itself to the peer.*

### Defining PAP

The Password Authentication Protocol (PAP) is used to verify that the PPP peer is a permitted device by checking a user name and password configured on the peer. The user name and password are both sent unencrypted across the connecting private circuit.

PAP requires two-way message passing. First, the router that is required to be authenticated (say the peer) sends an authentication request with its user name and password to the router requiring authentication (say the local router). The local router then looks up the user name and password in the user name database within the PPP interface, and if they match sends an authentication acknowledge back to the peer.

*The PPP user name and password database is separate and distinct from the global user name password database. For PAP and CHAP, use the database under the PPP interface configuration.*

Several example scenarios are given below for clarity.

**Configuring PAP Example 1: Only the local router requires the peer to authenticate itself.**

On the local router (host name **Local**):

Local(config-ppp 1)**#ppp authentication pap**
Local(config-ppp 1)**#username farend password far**

On the peer (host name **Peer**):

Peer(config-ppp 1)**#ppp pap sent-username farend password far**

The first line of the configuration sets the authentication mode as PAP. This means the peer is required to authenticate itself to the local router via PAP. The second line is the user name and password expected to be sent from the peer. On the peer, the **ppp pap sent-username** command is used to specify the appropriate matching user name and password.

**Configuring PAP Example 2: Both routers require the peer to authenticate itself.**

On the local router (host name **Local**):

Local(config-ppp 1)**#ppp authentication pap**
Local(config-ppp 1)**#username farend password far**
Local(config-ppp 1)**#ppp pap sent-username nearend password near**

On the peer (host name **Peer**):

Peer(config-ppp 1)**#ppp authentication pap**
Peer(config-ppp 1)**#username nearend password near**
Peer(config-ppp 1)**#ppp pap sent-username farend password far**

Now both routers send the authentication request, verify that the user name and password sent match what is expected in the database, and send an authentication acknowledge.

**Defining CHAP**

The Challenge-Handshake Authentication Protocol (CHAP) is a three-way authentication protocol composed of a challenge response and success or failure. The MD5 protocol is used to protect user names and passwords in the response.

First, the local router (requiring its peer to be authenticated) sends a “challenge” containing the unencrypted user name of the peer and a random number. The user name of the peer is found in the user name database within the PPP interface of the local router. The peer then looks up the user name in the user name database within the PPP interface, and if found takes the corresponding password and its own host name and sends a “response” back to the local router. This data is encrypted. The local router verifies that the user name and password are in its own user name database within the PPP interface, and if so sends a “success” back to the peer.

*The PPP user name and password database is separate and distinct from the global user name password database. For PAP and CHAP, use the database under the PPP interface configuration.*

Several example scenarios are given below for clarity.

**Configuring CHAP Example 1: Only the local router requires the peer to authenticate itself.**

On the local router (host name **Local**):

Local(config-ppp 1)**#ppp authentication chap**
Local(config-ppp 1)**#username Peer password same**

On the peer (host name **Peer**):

Peer(config-ppp 1)**#ppp chap password same**

The first line of this configuration sets the authentication mode to CHAP. This means the peer is required to authenticate itself to the local router via CHAP. The second line is the user name and password expected to be sent from the peer. The peer uses its **hostname** and **ppp chap password** commands to send the proper authentication information.

*Both ends must have identical passwords.*

**Configuring CHAP Example 2: Using the ppp chap hostname command as an alternate solution.**

On the local router (host name **Local**):

Local(config-ppp 1)**#ppp authentication chap**
Local(config-ppp 1)**#username farend password same**

On the peer (host name **Peer**):

Peer(config-ppp 1)**#ppp chap hostname farend**
Peer(config-ppp 1)**#ppp chap password same**

Notice the local router is expecting user name **farend** even though the peer router's host name is **Peer**. Therefore the peer router can use the **ppp chap hostname** command to send the correct name in the challenge.

*Both ends must have identical passwords.*

**Configuring CHAP Example 3: Both routers require each other to authenticate themselves using the same shared password.**

On the local router (host name **Local**):

Local(config-ppp 1)**#ppp authentication chap**
Local(config-ppp 1)**#username Peer password same**

On the peer (host name **Peer**):

Peer(config-ppp 1)**#ppp authentication chap**
Peer(config-ppp 1)**#username Local password same**

This is basically identical to Example 1 except that both routers will now challenge each other and respond.

*Both ends must have identical passwords.*

**Configuring CHAP Example 4: Both routers require each other to authenticate themselves using two separate shared passwords.**

On the local router (host name **Local**):

Local(config-ppp 1)**#ppp authentication chap**
Local(config-ppp 1)**#username Peer password far**
Local(config-ppp 1)**#ppp chap password near**

On the peer (host name **Peer**):

Peer(config-ppp 1)**#ppp authentication chap**
Peer(config-ppp 1)**#username Local password near**
Peer(config-ppp 1)**#ppp chap password far**

This is basically identical to Example 3, except that there are two separate shared passwords.

*Notice this example has both ends using different sets of passwords.*

**Configuring CHAP Example 5: Using the ppp chap hostname command as an alternate solution.**

On the local router (host name **Local**):

Local(config-ppp 1)**#ppp authentication chap**
Local(config-ppp 1)**#username farend password far**
Local(config-ppp 1)**#ppp chap hostname nearend**
Local(config-ppp 1)**#ppp chap password near**

On the peer (host name **Peer**):

Peer(config-ppp 1)**#ppp authentication chap**
Peer(config-ppp 1)**#username nearend password near**
Peer(config-ppp 1)**#ppp chap hostname farend**
Peer(config-ppp 1)**#ppp chap password far**

Notice the local router is expecting user name **farend** even though the peer router's host name is **Peer**. Therefore the peer router can use the **ppp chap hostname** command to send the correct name on the challenge.


*Notice this example has both ends using different sets of passwords.*

# ppp chap hostname *<name>*

Use the **ppp chap hostname** command to configure an alternate host name for CHAP PPP authentication. Use the **no** form of this command to remove a configured host name. For more information on PAP and CHAP functionality, refer to the *Technology Review* section for the command *ppp authentication* on page 1449.

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies a host name using an alphanumeric string up to 80 characters in length. |

## Default Values

By default, there are no configured PPP CHAP host names.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example specifies a PPP CHAP host name of **my_host**:

(config)**#interface ppp 1**
(config-ppp 1)**#ppp chap hostname my_host**

# ppp chap password *<password>*

Use the **ppp chap password** command to configure an alternate password when the peer requires CHAP PPP authentication. Use the **no** form of this command to remove a configured password. For more information on PAP and CHAP functionality, refer to the *Technology Review* section for the command *ppp authentication* on page 1449.

## Syntax Description

| | |
|---|---|
| *<password>* | Specifies a password using an alphanumeric string up to 80 characters in length. |

## Default Values

By default, there is no defined PPP CHAP password.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example specifies a PPP CHAP password of **my_password**:

(config)**#interface ppp 1**
**(config-ppp 1)#**ppp chap password my_password

# ppp multilink

Use the **ppp multilink** command to enable multilink PPP (MPPP) operation om an existing PPP interface. Use the **no** form of this command to disable. Variations of this command include:

**ppp multilink fragmentation**
**ppp multilink interleave**
**ppp multilink maximum** <*number*>

## Syntax Description

| | |
|---|---|
| **fragmentation** | Enables multilink fragmentation operation. |
| **interleave** | Enables multilink interleave operation. |
| **maximum** <*number*> | Specifies the maximum number of links allowed in a PPP multilink bundle. |

## Default Values

By default, MPPP is disabled.

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |
| Release 7.2 | Fragmentation and interleave operation were added. |
| Release 11.1 | Command expanded to include the demand interface. |

## Functional Notes

When enabled, this interface is capable of the following:

- Combining multiple physical links into one logical link.
- Receiving upper layer protocol data units (PDU), fragmenting and transmitting over the physical links.
- Receiving fragments over the physical links and reassembling them into PDUs.

The fragmentation and interleave options can be used to enhance the multilink operation. Fragmentation is used to reduce serialization delays of large packets. The fragmentation process evenly divides the data among all links in the bundle with a minimum packet size of 96 bytes. The interleave operation is used with streaming protocols to reduce delay by giving priority to packets identified as high priority. In order delivery is guaranteed with multilink fragmentation, but is not guaranteed with multilink interleave operation.

The multilink bundle will remain active with a minimum of one physical link. Physical links may be dynamically added or removed from the multilink bundle with minor interruption to traffic flow.

## Usage Examples

The following example enables MPPP:

(config)**#interface ppp 1**
(config-ppp 1)**#ppp multilink**

# ppp pap sent-username <*username*> password <*password*>

Use the **ppp pap sent-username password** command to configure a user name and password when the peer requires PAP PPP authentication. Use the **no** form of this command to remove a configured password. For more information on PAP and CHAP functionality, refer to the *Technology Review* section for the command *ppp authentication* on page 1449.

## Syntax Description

| | |
|---|---|
| *<username>* | Specifies a user name by alphanumeric string up to 80 characters in length (the user name is case-sensitive). |
| *<password>* | Specifies a password by alphanumeric string up to 80 characters in length (the password is case-sensitive). |

## Default Values

By default, there is no defined **ppp pap sent-username** and **password**.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example specifies a PPP PAP sent-user name of **local** and a password of **my_password**:

(config)**#interface ppp 1**
(config-ppp 1)**#ppp pap sent-username local password my_password**

# pppoe ac-name *<name>*

Use the **pppoe ac-name** command to identify the Access Concentrator (AC) with which the AOS expects to establish a PPPoE session. Use the **no** form of this command to return to the default setting.

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies an AC by text string (up to 255 characters) corresponding to the AC-Name Tag under RFC2516. If this field is not specified, any access concentrator is acceptable. The AC value may be a combination of trademark, model, and serial ID information (or simply the MAC address of the unit). |

## Default Values

By default, no AC is specified.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following example identifies the AC with which the AOS expects to establish a PPPoE session:

(config)**#interface ppp 1**
(config-ppp 1)**#pppoe acc-name Access_Concentrator_Name**

# pppoe service-name <*name*>

Use the **pppoe service-name** command to use this tag value to filter PPPoE session offers from PPPoE servers. Use the **no** form of this command to return to the default setting.

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies a service name by text string (up to 255 characters) corresponding to the Service-Name Tags under RFC2516. This string indicates an ISP name (or a class or quality of service). If this field is not specified, any service is acceptable. |

## Default Values

By default, no names are specified.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following example defines a service type that is not to be accepted by the AOS:

(config)**#interface ppp 1**
(config-ppp 1)**#pppoe service-name Service_Name**

# qos-policy out *<name>*

Use the **qos-policy out** command to apply a previously-configured QoS map to the outgoing packets on an interface. Use the **no** form of this command to remove the map from the interface.

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies the name of a previously-created QoS map (refer to *qos map <name> <number>* on page 643 for more information). |

## Default Values

No default value is necessary for this command.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Functional Notes

When a QoS policy is applied to an interface, it may be disabled if the interface bandwidth is not adequate to support the requested bandwidth on the map set. Once the bandwidth problem is resolved, the map will work again. The bandwidth will be rechecked on any of the following changes:

1. A priority or class-based entry is added to, deleted from, or changed in a QoS map set.
2. The interface bandwidth is changed by the **bandwidth** command on the interface.
3. A QoS policy is applied to an interface.
4. A cross-connect is created that includes an interface with a QoS policy.
5. The interface queuing method is changed to fair-queue to use weighted fair queuing.
6. The interface operational status changes.
7. The interface bandwidth changes for other reasons (e.g., when ADSL finishes training).

In order to prevent the map from being disabled in cases of temporary inadequate bandwidth (e.g., a single link goes down in a dual T1 multilink configuration where the map requests more than one T1's worth of bandwidth), the QoS map uses the maximum theoretical bandwidth on an interface, not the actual bandwidth at that time. This actually helps QoS keep higher priority class-based traffic working better than best-effort traffic when the bandwidth drops.

## Usage Examples

The following example applies the QoS map **VOICEMAP** to the PPP 1 interface:

(config)**#interface ppp 1**
(config-ppp 1)**#qos-policy out VOICEMAP**

# snmp trap link-status

Use the **snmp trap link-status** command to control the Simple Network Management Protocol (SNMP) variable ifLinkUpDownTrapEnable (RFC2863) to enable (or disable) the interface to send SNMP traps when there is an interface status change. Use the **no** form of this command to disable this trap.

## Syntax Description

No subcommands.

## Default Values

By default, the ifLinkUpDownTrapEnable OID is enabled for all interfaces except virtual Frame Relay interfaces.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 3.1 | Command was extended to the SHDSL interface. |
| Release 5.1 | Command was expanded to include Gigabit Ethernet, port channel, VLAN, E1, and G.703 interfaces. |

## Functional Notes

The **snmp trap link-status** command is used to control the RFC2863 ifLinkUpDownTrapEnable OID (OID number 1.3.6.1.2.1.31.1.1.1.14.0).

## Usage Examples

The following example disables the link-status trap on the virtual PPP interface:

(config)**#interface ppp 1**
(config-ppp 1)**#no snmp trap link-status**

# **username** *<username>* **password** *<password>*

Use the **username password** command to configure the user name and password of the peer to use for PPP authentication. Use the **no** form of this command to remove a configured user name and password.

## Syntax Description

| | |
|---|---|
| *<username>* | Specifies a user name by alphanumerical string up to 30 characters in length (the user name is case-sensitive). |
| *<password>* | Specifies a password by alphanumerical string up to 30 characters in length (the password is case-sensitive). |

## Default Values

By default, there is no established user name and password.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

PAP uses this entry to check received information from the peer. CHAP uses this entry to check the received peer host name and a common password.

## Usage Examples

The following example creates a user name of **ADTRAN** with password **ADTRAN** for the PPP link labeled 5:

(config)**#interface ppp 5**
(config-ppp 5)**#username ADTRAN password ADTRAN**

# TUNNEL CONFIGURATION COMMAND SET

To activate the Tunnel Configuration mode, enter the **interface tunnel** command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#interface tunnel 1
(config-tunnel 1)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:



All other commands for this command set are described in this section in alphabetical order.



# access-policy *<name>*

Use the **access-policy** command to assign a specified access policy to an interface. Access policies are applied to traffic entering an interface. Use the **no** form of this command to remove an access policy association. For more information on using access policies, refer to *ip policy-class <name>* on page 549.

> NOTE
>
> *Configured access policies will only be active if the **ip firewall** command has been entered at the Global Configuration mode prompt to enable the AOS security features. All configuration parameters are valid, but no security data processing will be attempted unless the security features are enabled.*

## Syntax Description

| | |
|---|---|
| *<name>* | Identifies the configured access policy by alphanumeric descriptor (all access policy descriptors are case-sensitive). |

## Default Values

By default, there are no configured access policies associated with an interface.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include Ethernet sub-interfaces. |
| Release 6.1 | Command was expanded to include VLAN interfaces. |

## Functional Notes

To assign an access policy to an interface, enter the interface configuration mode for the desired interface and enter **access-policy** *<policy name>*.

## Usage Examples

The following example associates the access policy **Private** (to allow inbound traffic to the Web server) to the tunnel interface 1:

Enable the AOS security features:
(config)**#ip firewall**

Associate the access policy with the tunnel interface 1:

(config)**#interface tunnel 1**
(config-tunnel 1)**#access-policy Private**

# bandwidth *<value>*

Use the **bandwidth** command to provide the bandwidth value of an interface to the higher-level protocols. This value is used in cost calculations. Use the **no** form of this command to restore the default values.

## Syntax Description

| | |
|---|---|
| *<value>* | Specifies bandwidth in kbps. Range is 1 to 4,294,967,295 kbps. |

## Default Values

To view default values, use the **show interfaces** command.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include Ethernet sub-interfaces. |
| Release 6.1 | Command was expanded to include VLAN interfaces. |
| Release 9.1 | Command was expanded to include tunnel interfaces. |

## Functional Notes

The **bandwidth** command is an informational value that is communicated to the higher-level protocols to be used in cost calculations. This is a routing parameter only and does not affect the physical interface.

## Usage Examples

The following example sets bandwidth of the tunnel 1 interface to 10 Mbps:

(config)**#interface tunnel 1**
(config-tunnel 1)**#bandwidth 10000**

# crypto map *<name>*

Use the **crypto map** command to associate crypto maps with the interface. Use the **no** form of this command to remove a crypto map from an interface.

> NOTE *When you apply a map to an interface, you are applying all crypto maps with the given map name. This allows you to apply multiple crypto maps if you have created maps which share the same name but have different map index numbers.*

> NOTE *For VPN configuration example scripts, refer to the technical support note **Configuring VPN** located on the ADTRAN OS System Documentation CD provided with your unit.*

## Syntax Description

| | |
|---|---|
| *<name>* | Assigns a crypto map name to the interface. |

## Default Values

By default, no crypto maps are assigned to an interface.

## Command History

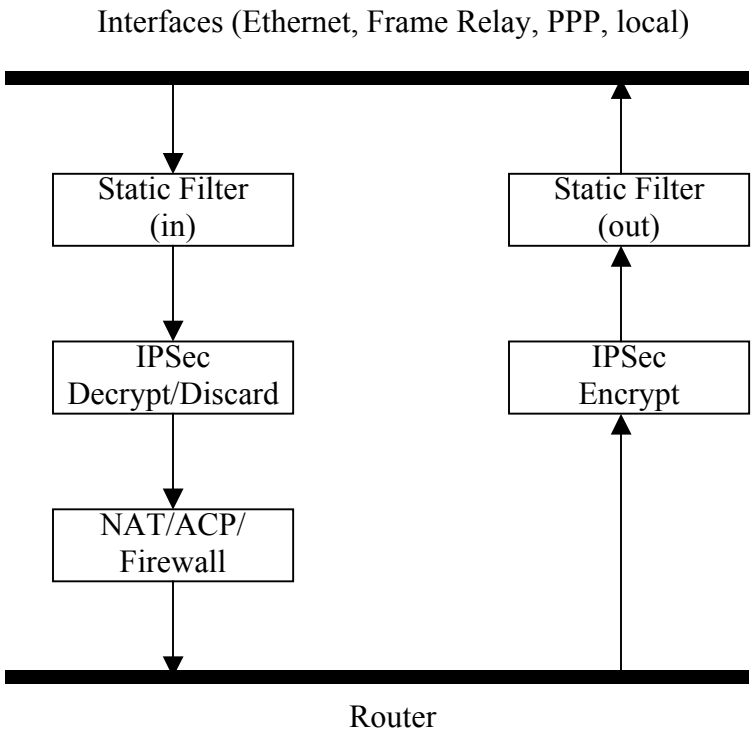
| | |
|---|---|
| Release 4.1 | Command was introduced. |

## Functional Notes

When configuring a system to use both the stateful inspection firewall and IKE negotiation for VPN, keep the following notes in mind.

When defining the policy class and associated access-control lists (ACLs) that describe the behavior of the firewall, do not forget to include the traffic coming into the system over a VPN tunnel terminated by the system. The firewall should be set up with respect to the unencrypted traffic that is destined to be sent or received over the VPN tunnel. The following diagram represents typical AOS data-flow logic.




As shown in the diagram above, data coming into the product is first processed by the static filter associated with the interface on which the data is received. This access group is a true static filter and is available for use regardless of whether the firewall is enabled or disabled. Next (if the data is encrypted) it is sent to the IPSec engine for decryption. The decrypted data is then processed by the stateful inspection firewall. Therefore, given a terminating VPN tunnel, only unencrypted data is processed by the firewall.

The ACLs for a crypto map on an interface work in reverse logic to the ACLs for a policy class on an interface. When specifying the ACLs for a crypto map, the source information is the private local-side, unencrypted source of the data. The destination information will be the far end, unencrypted destination of the data. However, ACLs for a policy class work in reverse. The source information for the ACL in a policy class is the far end. The destination information is the local side.

## Usage Examples

The following example applies all crypto maps with the name **MyMap** to the tunnel interface:

(config)**#interface tunnel 1**
(config-tunnel 1)**#crypto map MyMap**

# dial-backup auto-backup

Use the **dial-backup auto-backup** command to configure the interface to automatically attempt a dial-backup upon failure. Use the **no** form of this command to disable this feature. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of *dial-backup call-mode* on page 1471.

## Syntax Description

No subcommands.

## Default Values

By default, all backup endpoints will automatically attempt dial-backup upon a failure.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include the PPP interface. |

## Usage Examples

The following example enables automatic dial-backup on the endpoint:

(config)**#interface tunnel 1**
(config-tunnel 1)**#dial-backup auto-backup**

# dial-backup auto-restore

Use the **dial-backup auto-restore** command to configure the interface to automatically discontinue dial-backup when all network conditions are operational. Use the **no** form of this command to disable the auto-restore feature. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of *dial-backup call-mode* on page 1471.

## Syntax Description

No subcommands.

## Default Values

By default, all backup endpoints will automatically restore the primary connection when the failure condition clears.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include the PPP interface. |

## Usage Examples

The following example configures the AOS to automatically restore the primary connection when the failure condition clears:

(config)**#interface tunnel 1**
(config-tunnel 1)**#dial-backup auto-restore**

# dial-backup backup-delay *<value>*

Use the **dial-backup backup-delay** command to configure the amount of time the router will wait after the failure condition is recognized before attempting to backup the link. Use the **no** form of this command to return to the default value. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of *dial-backup call-mode* on page 1471.

## Syntax Description

| | |
|---|---|
| *<value>* | Specifies the delay period (in seconds) a failure must be active before the AOS will enter backup operation on the interface. Range is 10 to 86,400 seconds. |

## Default Values

By default, the **dial-backup backup-delay** period is set to 10 seconds.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include the PPP interface. |

## Usage Examples

The following example configures the AOS to wait 60 seconds (on an endpoint with an active alarm condition) before attempting dial-backup operation:

(config)**#interface tunnel 1**
(config-tunnel 1)**#dial-backup backup-delay 60**

# dial-backup call-mode

Use the **dial-backup call-mode** command to specify whether the configured backup interface answers or originates (or a combination of both ) backup calls. Use the **no** form of this command to return to the default value. Variations of this command include:

**dial-backup call-mode answer**
**dial-backup call-mode answer-always**
**dial-backup call-mode originate**
**dial-backup call-mode originate-answer**
**dial-backup call-mode originate-answer-always**

## Syntax Description

| | |
|---|---|
| **answer** | Answers and backs up primary link on failure. |
| **answer-always** | Answers and backs up regardless of primary link state. |
| **originate** | Originates backup call on primary link failure. |
| **originate-answer** | Originates or answers call on primary link failure. |
| **originate-answer-always** | Originates on failure; answers and backs up always. |

## Default Values

By default, the **dial-backup call-mode** is set to **originate-answer**.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include the PPP interface. |

## Functional Notes

The majority of the configuration for AOS dial-backup implementation is configured via the dial-backup PPP interface configuration commands. However, the numbers dialed are configured in the primary interface. Full sample configurations follow:

**Sample configuration for remote router (dialing out)**

```
hostname “Remote3200”
enable password adtran
!
interface eth 0/1
 ip address 192.168.1.254 255.255.255.0
 no shutdown
!
interface modem 1/3
 no shutdown
!
interface t1 1/1
```

```
 coding b8zs
 framing esf
 clock source line
 tdm-group 1 timeslots 1-24
 no shutdown
!
interface fr 1 point-to-point
 frame-relay lmi-type ansi
 no shutdown
 cross-connect 1 t1 1/1 1 fr 1
!
interface fr 1.16 point-to-point
 frame-relay interface-dlci 16
 ip address 10.1.1.2 255.255.255.252
 dial-backup call-mode originate
 dial-backup number 5551111 analog ppp1
 dial-backup number 5552222 analog ppp1
  no shutdown
!
interface ppp 1
 ip address 172.22.56.1 255.255.255.252
 ppp authentication chap
 username remoterouter password remotepass
 ppp chap hostname localrouter
 ppp chap password adtran
 no shutdown
!
ip route 192.168.2.0 255.255.255.0 172.22.56.2 255.255.255.252
!
line telnet 0 4
 password password
```

**Sample configuration for central router (dialing in)**

```
hostname “Central3200”
enable password adtran
!
interface eth 0/1
 ip address 192.168.100.254 255.255.255.0
 no shutdown
!
interface modem 1/3
 no shutdown
!
interface t1 1/1
```

```
 coding b8zs
 framing esf
 clock source line
 tdm-group 1 timeslots 1-24
 no shutdown
!
interface fr 1 point-to-point
 frame-relay lmi-type ansi
 no shutdown
 cross-connect 1 t1 1/1 1 fr 1
!
interface fr 1.100 point-to-point
 frame-relay interface-dlci 100
 ip address 10.1.1.1 255.255.255.252
 dial-backup call-mode answer
 dial-backup number 555-8888 analog ppp 1
!
interface ppp 1
 ip address 172.22.56.2 255.255.255.252
 ppp authentication chap
 username localrouter password adtran
 ppp chap hostname remoterouter
 ppp chap password remotepass
 no shutdown
!
ip route 192.168.1.0 255.255.255.0 172.22.56.1 255.255.255.252

line telnet 0 4
 password password
```

## Usage Examples

The following example configures AOS to generate backup calls for this endpoint using an analog modem interface (to phone number 555 1111) but never answer calls and specifies **ppp 2** as the backup interface:

```
(config)#interface tunnel 1
(config-tunnel 1)#backup call-mode originate
(config-tunnel 1)#backup number 555 1111 analog ppp 2
```

## Technology Review

This technology review provides information regarding specific dial-backup router behavior (i.e., when the router will perform dial-backup, where in the configuration the AOS accesses specific routing information, etc.):

### Dialing Out

1. The AOS determines to place an outbound call when either the Layer 1 or Layer 2 has a failure.
2. When placing outbound calls, the AOS matches the number dialed to a PPP interface. This is accomplished with an addition to the **dial-backup number** command (refer to *dial-backup number* on page 1478).
3. When placing the call, the AOS uses the configuration of the related PPP interface for authentication and IP negotiation.
4. If the call fails to connect on the first number dialed, the AOS places a call to the second number (if a second number is configured). The second number to be dialed references a separate PPP interface.

### Dialing In

1. The AOS receives an inbound call on a physical interface.
2. Caller ID is used to match the **dial-backup number** command to the configured PPP interface.
3. If a match is found, the call connects and the AOS pulls down the primary connection if it is not already in a down state.
4. If no match is found from caller ID, the call is terminated.

# dial-backup connect-timeout <*value*>

Use the **dial-backup connect-timeout** command to specify the number of seconds to wait for a connection after a call is attempted before trying to call again or dialing a different number. It is recommended this number be greater than 60. Use the **no** form of this command to return to the default setting. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of *dial-backup call-mode* on page 1471.

## Syntax Description

| | |
|---|---|
| <*value*> | Selects the amount of time (in seconds) that the router will wait for a connection before attempting another call. Valid range is 10 to 300 seconds. |

## Default Values

By default, the **dial-backup connect-timeout** period is set to 60 seconds.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include the PPP interface. |

## Usage Examples

The following example configures the AOS to wait 120 seconds before retrying a failed dial-backup call:

(config)**#interface tunnel 1**
(config-tunnel 1)**#dial-backup connect-timeout 120**

# dial-backup force

Use the **dial-backup force** command to manually override the automatic dial-backup feature. This can be used to force a link into backup to allow maintenance to be performed on the primary link without disrupting data. Use the **no** form of this command to return to the normal dial-backup operation state. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of *dial-backup call-mode* on page 1471. Variations of this command include:

**dial-backup force backup**
**dial-backup force primary**

## Syntax Description

| | |
|---|---|
| **backup** | Force backup regardless of primary link state. |
| **primary** | Force primary link regardless of its state. |

## Default Values

By default, this feature is disabled.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include the PPP interface. |

## Usage Examples

The following example configures the AOS to force this interface into dial-backup:

(config)**#interface tunnel 1**
(config-tunnel 1)**#dial-backup force backup**

# dial-backup maximum-retry <*value*>

Use the **dial-backup maximum-retry** command to select the number of calls the router will make when attempting to backup a link. Use the **no** form of this command to return to the default state. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of *dial-backup call-mode* on page 1471.

## Syntax Description

| | |
|---|---|
| <*value*> | Selects the number of call retry attempts that will be made after a link failure. Valid range is 0 to 15 attempts.<br>Setting this value to 0 will allow unlimited retries during the time the network is failed. |

## Default Values

By default, **dial-backup maximum-retry** is set to 0 attempts.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include the PPP interface. |

## Usage Examples

The following example configures the AOS to retry a dial-backup call four times before considering backup operation not available:

(config)**#interface tunnel 1**
(config-tunnel 1)**#dial-backup maximum-retry 4**

# dial-backup number

Use the **dial-backup number** command to configure the phone number and the call type the router will dial upon network failure. Multiple entries can be made for an interface to allow alternate sites to be dialed. Use the **no** form of this command to return to the default setting. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of the command *dial-backup call-mode* on page 1471. Variations of this command include:

**dial-backup number** <*number*> **analog ppp** <*interface*>
**dial-backup number** <*number*> **digital-56k** <*isdn min chan*> <*isdn max chan*> **ppp** <*interface*>
**dial-backup number** <*number*> **digital 64k** <*isdn min chan*> <*isdn max chan*> **ppp** <*interface*>

## Syntax Description

| | |
|---|---|
| <*number*> | Specifies the phone numbers to call when the backup is initiated. |
| **analog ppp** | Indicates the number connects to an analog modem. |
| **digital-56k** | Indicates the number belongs to a digital 56 kbps per DS0 connection. |
| **digital-64k** | Indicates the number belongs to a digital 64 kbps per DS0 connection. |
| <*isdn min chan*> | Specifies the minimum number of DS0s required for a digital 56 or 64 kbps connection. Range is 1 to 24. |
| <*isdn max chan*> | Specifies the maximum number of DS0s desired for a digital 56 or 64 kbps connection. Range is 1 to 24. |
| **ppp** <*interface*> | Specifies the PPP interface to use as the backup for this interface. |

## Default Values

By default, there are no configured dial-backup numbers.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include the PPP interface. |

## Usage Examples

The following example configures the AOS to dial 704-555-1212 (digital 64 kbps connection) to initiate dial-backup operation on this endpoint using interface **ppp 1** backup interface:

(config)**#interface tunnel 1**
(config-tunnel 1)**#dial-backup number 7045551212 digital-64k 1 1 ppp 1**

# dial-backup priority <*value*>

Use the **dial-backup priority** command to select the backup priority for this interface. This allows the user to establish the highest priority backup link and ensure that link will override backups attempted by lower priority links. Use the **no** form of this command to return to the default value. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of the command *dial-backup call-mode* on page 1471.

## Syntax Description

| | |
|---|---|
| <*value*> | Sets the relative priority of this link. Valid range is 0 to 100. A value of 100 designates the highest priority. |

## Default Values

By default, **dial-backup priority** is set to 50.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include the PPP interface. |

## Usage Examples

The following example assigns the highest priority to this endpoint:

(config)**#interface tunnel 1**
(config-tunnel 1)**#dial-backup priority 100**

# dial-backup randomize-timers

Use the **dial-backup randomize-timers** command to randomize the call timers to minimize potential contention for resources. Use the **no** form of this command to return to the default value. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of the command *dial-backup call-mode* on page 1471.

## Syntax Description

No subcommands.

## Default Values

By default, the AOS does not randomize the dial-backup call timers.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include the PPP interface. |

## Usage Examples

The following example configures the AOS to randomize the dial-backup timers associated with this endpoint:

(config)**#interface tunnel 1**
(config-tunnel 1)**#dial-backup randomize-timers**

# dial-backup redial-delay *<value>*

Use the **dial-backup redial-delay** command to configure the delay after an unsuccessful call until the call will be re-tried. Use the **no** form of this command to return to the default setting. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of the command *dial-backup call-mode* on page 1471.

## Syntax Description

| | |
|---|---|
| *<value>* | Specifies the delay in seconds between attempting to re-dial a failed backup attempt. Range is 10 to 3600 seconds. |

## Default Values

By default, **dial-backup redial-delay** is set to 10 seconds.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include the PPP interface. |

## Usage Examples

The following example configures a redial delay of **25** seconds on this endpoint:

(config)**#interface tunnel 1**
(config-tunnel 1)**#dial-backup redial-delay 25**

# dial-backup restore-delay *<value>*

Use the **dial-backup restore-delay** command to configure the amount of time the router will wait after the network is restored before disconnecting the backup link and reverting to the primary. This setting is used to prevent disconnecting the backup link if the primary link is "bouncing" in and out of alarm. Use the **no** form of this command to return to the default setting. For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of the command *dial-backup call-mode* on page 1471.

## Syntax Description

| | |
|---|---|
| *<value>* | Specifies the number of seconds the AOS will wait (after a primary link is restored) before disconnecting dial-backup operation. Range is 10 to 86,400 seconds. |

## Default Values

By default, **dial-backup restore-delay** is set to 10 seconds.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include the PPP interface. |

## Usage Examples

The following example configures the AOS to wait **30** seconds before disconnecting dial-backup operation and restoring the primary connection for this endpoint:

(config)**#interface tunnel 1**
(config-tunnel 1)**#dial-backup restore-delay 30**

# dial-backup schedule

Use the **dial-backup schedule** command to set the time of day that backup will be enabled. Use this command if backup is desired only during normal business hours and on specific days of the week. Use the **no** form of this command to disable dial-backup (as specified). For more detailed information on dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of the command *dial-backup call-mode* on page 1471.Variations of this command include:

**dial-backup schedule day** <*name*>
**dial-backup schedule enable-time** <*value*>
**dial-backup schedule disable-time** <*value*>

## Syntax Description

| | |
|---|---|
| **day** <*name*> | Sets the days to allow backup. Valid range is Monday through Sunday. |
| **enable-time** <*value*> | Sets the time of day to enable backup. Time is entered in 24-hour format (00:00). |
| **disable-time** <*value*> | Sets the time of day to disable backup. Time is entered in 24-hour format (00:00). |

## Default Values

By default, dial-backup is enabled for all days and times if the dial-backup auto-backup command has been issued and the dial-backup schedule has not been entered.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include the PPP interface. |

## Usage Examples

The following example enables dial-backup Monday through Friday 8:00 am to 7:00 pm:

(config)**#interface tunnel 1**
(config-tunnel 1)**#dial-backup schedule enable-time 08:00**
(config-tunnel 1)**#dial-backup schedule disable-time 19:00**
(config-tunnel 1)**#no dial-backup schedule day Saturday**
(config-tunnel 1)**#no dial-backup schedule day Sunday**

# dial-backup shutdown

Use the **dial-backup shutdown** command to deactivate all dial-backup functionality in the unit. Dial-backup configuration parameters are kept intact, but the unit will not initiate (or respond) to dial-backup sequences in the event of a network outage. Use the **no** form of this command to reactivate the dial-backup interface. For more detailed information on PPP dial-backup functionality, refer to the *Functional Notes* and *Technology Review* sections of the command.

## Syntax Description

No subcommands.

## Default Values

By default, all AOS interfaces are disabled.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include the PPP interface. |

## Usage Examples

The following example deactivates the configured dial-backup interface:

(config)**#interface tunnel 1**
(config-tunnel 1)**#dial-backup shutdown**

# dynamic-dns

Use the **dynamic-dns** command to configure Dynamic DNS service provided by Dynamic Network Services, Inc. (www.dyndns.org). Use the **no** versions of these commands to disable these features. Variations of this command include:

**dynamic-dns custom** <*hostname*> <*minutes*>
**dynamic-dns dyndns** <*hostname*> <*username*> <*password*>
**dynamic-dns dyndns-custom** <*hostname*> <*username*> <*password*>
**dynamic-dns dyndns-static** <*hostname*> <*username*> <*password*>

## Syntax Description

| | |
|---|---|
| <*hostname*> | Specifies the host name for the server that updates the Dynamic Domain Name Server (DNS). |
| <*minutes*> | Specifies the intervals in minutes to update the server with information (updates also occur when the interface's IP address changes regardless of the update intervals).<br>Refer to *Functional Notes* below for additional argument descriptions. |

## Default Values

No default is necessary for this command.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |
| Release 12.1 | Command was expanded. |

## Functional Notes

**custom** - Constanttime.com's Custom Dynamic DNS$^{SM}$ service allows you complete access and management control over your domain name regardless of where you purchased/registered it. This allows to manage IP address mappings (A records), domain aliases (CNAME records) and mail servers (MX records).

**dyndns** - The Dynamic DNS$^{SM}$ service allows you to alias a dynamic IP address to a static host name in various domains. This allows your unit to be more easily accessed from various locations on the Internet. This service is provided for up to five host names.

**dyndns-custom -** DynDNS.org's Custom DNS$^{SM}$ service provides a full DNS solution, giving you complete control over an entire domain name. A web-based interface provides two levels of control over your domain, catering to average or advanced users. Five globally redundant DNS servers ensure that your domain will always resolve.

A choice of two interfaces is available. The basic interface is designed for most users. It comes preconfigured for the most common configuration and allows for easy creation of most common record types. The advanced interface is designed for system administrators with a solid DNS background, and provides layout and functionality similar to a BIND zone file allowing for the creation of nearly any record type.

Custom DNS$^{SM}$ can be used with both static and dynamic IPs and has the same automatic update capability through Custom DNS-aware clients as Dynamic DNS.

**dyndns-static -** The Static DNS service is similar to Dynamic DNS service in that it allows a host name such as yourname.dyndns.org to point to your IP address. Unlike a Dynamic DNS host, a Static DNS host does not expire after 35 days without updates, but updates take longer to propagate though the DNS system. This service is provided for up to five host names.

If your IP address does not change often or at all but you still want an easy name to remember it by (without having to purchase your own domain name), Static DNS service is ideal for you.

If you would like to use your own domain name (such as yourname.com), you need Custom DNS service which also provides full dynamic and static IP address support.

**Usage Examples**

The following example sets the dynamic-dns to **dyndns-custom** with host name **host**, user name **user**, and password **pass**:

(config)**#interface tunnel 1**
(config-tunnel 1)**#dynamic-dns dyndns-custom host user pass**

# ip access-group *<name>*

Use the **ip access-group** command to create an access control list to be used for packets transmitted on or received from the specified interface. Use the **no** form of this command to disable this type of control. Variations of this command include:

**ip access-group** <*name*> **in**
**ip access-group** <*name*> **out**

## Syntax Description

| | |
|---|---|
| *<name>* | Assigns an IP access control list name. |
| **in** | Enables access control on packets received on the specified interface. |
| **out** | Enables access control on packets transmitted on the specified interface. |

## Default Values

By default, these commands are disabled.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |
| Release 9.1 | Command was expanded to include tunnel interfaces. |

## Functional Notes

When this command is enabled, the IP destination address of each packet must be validated before being passed through. If the packet is not acceptable per these settings, it is dropped.

## Usage Examples

The following example sets up the unit to only allow Telnet traffic (as defined in the user-configured **TelnetOnly** IP access control list) into the tunnel interface:

(config)**#ip access-list extended TelnetOnly**
(config-ext-nacl)**#permit tcp any any eq telnet**
(config-ext-nacl)**#interface tunnel 1**
(config-tunnel 1)**#ip access-group TelnetOnly in**

# ip address *<ip address> <subnet mask>*

Use the **ip address** command to define an IP address on the specified interface. Use the **no** form of this command to remove a configured IP address. Variations of this command include:

**ip address** *<ip address> <subnet mask>*
**ip address** *<ip address> <subnet mask>* **secondary**

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies a valid IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| *<subnet mask>* | Specifies the subnet mask that corresponds to a range of IP addresses (network) or a specific host. Subnet masks can be expressed in dotted decimal notation (for example, **255.255.255.0**) or as a prefix length (for example, **/24**). |
| **secondary** | Optional. Configures a secondary IP address for the specified interface. |

## Syntax Description

| | |
|---|---|
| *<address>* | Defines the IP address for the interface in dotted decimal notation (for example: 192.22.73.101). |
| *<mask>* | Specifies the subnet mask that corresponds to the listed IP address. |
| **secondary** | Optional. Configures a secondary IP address for the specified interface. |

## Default Values

By default, there are no assigned IP addresses.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |
| Release 9.1 | Command was expanded to include tunnel interfaces. |

## Usage Examples

The following example configures an IP address of **192.22.72.101/30**:

(config)**#interface tunnel 1**
(config-tunnel 1)**#ip address 192.22.72.101 255.255.255.252**

# ip ffe

Use the **ip ffe** command to enable the FastFlow Engine (FFE) on this interface with the default number of entries. Use the **no** form of these commands to disable the FFE. Variations of this command include:

**ip ffe max-entries** <*entries*>

*Issuing this command will cause all FFE entries on this interface to be cleared.*

## Syntax Description

| | |
|---|---|
| **max-entries** <*entries*> | Specifies the maximum number of entries stored in the flow table. Valid range is from 1 to 8192. |

## Default Values

By default, the FFE is disabled. The default number of **max-entries** is 4096.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Functional Notes

The FFE can be used to help reduce routing overhead and thus reduce overall routing times. Routing times are reduced by the creation of a flow table on the ingress interface. The maximum number of entries that can be stored in the flow table at any one time may be specified by using the **max-entries** subcommand. When features such as firewall, VPN, and static filters are enabled, routing time reduction may not be realized.

## Usage Examples

The following example enables the FFE and sets the maximum number of entries in the flow table to 50:

(config)**#interface tunnel 1**
(config-tunnel 1)**#ip ffe max-entries 50**

## Technology Review

The FastFlow system goal is to increase IP packet throughput by moving as much of the packet processing into the engine as possible. Packets are classified into flows based upon the IP protocol (TCP, UDP, ICMP, etc.), the source and destination IP addresses, IP TOS, and the protocol-specific information such as the source and destination port numbers. Flows are defined as the unidirectional representation of a conversation between two IP hosts. Each ingress interface keeps its own flow table, a collection of flow entries.

The first packet in a flow that is forwarded through the unit will build a flow entry. When a flow entry is looked up but no entry is found, a FastFlowBuilder object is allocated and attached to the packet. As the packet passes through the various processing layers, each subsystem will add processing to the FastFlowBuilder. When packet is about to be forwarded out of the egress interface, the FastFlowBuilder will be finalized. That is, the flow entry being built will be checked for completeness and committed to the flow table on the ingress interface. Subsequent flow matches can then bypass the normal processing layers.

# ip helper-address *<ip address>*

Use the **ip helper-address** command to configure the AOS to forward User Datagram Protocol (UDP) broadcast packets received on the interface. Use the **no** form of this command to disable forwarding packets.

*The **ip helper-address** command must be used in conjunction with the **ip forward-protocol** command to configure the AOS to forward UDP broadcast packets. Refer to ip forward-protocol udp <value> on page 532 for more information.*

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies the destination IP address for the forwarded UDP packets. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

## Default Values

By default, broadcast UDP packets are not forwarded.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 9.1 | Command was expanded to include tunnel interfaces. |

## Functional Notes

When used in conjunction with the **ip forward-protocol** command, the **ip helper-address** feature allows you to customize which broadcast packets are forwarded.

To implement the helper address feature, assign helper address(es) (specifying the device that needs to receive the broadcast traffic) to the interface closest to the host that transmits the broadcast packets. When broadcast packets (of the specified type forwarded using the **ip forward-protocol** command) are received on the interface, they will be forwarded to the device that needs the information.

Only packets meeting the following criteria are considered eligible by the **ip helper-address** feature:

1. The packet IP protocol is User Datagram Protocol (UDP).
2. Any UDP port specified using the **ip forward-protocol** command.
3. The media access control (MAC) address of the frame is an all-ones broadcast address (ffff.ffff.ffff).
4. The destination IP address is broadcast defined by all ones (255.255.255.255) or a subnet broadcast (for example, 192.33.4.251 for the 192.33.4.248/30 subnet).

## Usage Examples

The following example forwards all DNS broadcast traffic to the DNS server with IP address 192.33.5.99:

(config)**#ip forward-protocol udp domain**
(config)**#interface tunnel 1**
(config-tunnel 1)**#ip helper-address 192.33.5.99**

# ip igmp

Use the **ip igmp** command to configure multicasting-related functions for the interface.Variations of this command include:

**ip igmp immediate-leave**
**ip igmp last-member-query-interval** <*milliseconds*>
**ip igmp querier-timeout** <*seconds*>
**ip igmp query-interval** <*seconds*>
**ip igmp query-max-response-time** <*seconds*>
**ip igmp static-group** <*address*>
**ip igmp version [1 | 2]**

## Syntax Description

| | |
|---|---|
| **immediate-leave** | Specifies that if only one host (or IGMP snooping switch) is connected to the interface, when a leave is received, multicast of that group is immediately terminated as opposed to sending a group query and timing out the group if no device responds. Works in conjunction with **ip igmp last-member-query-interval**. Applies to all groups when configured. Use the **no** form of this command to disable the **immediate-leave** feature. |
| **last-member-query-interval** <*milliseconds*> | Controls the timeout (in milliseconds) used to detect whether any group receivers remain on an interface after a receiver leaves a group. If a receiver sends a leave-group message (IGMP Version 2), the router sends a group-specific query on that interface. After twice the time specified by this command plus as much as one second longer, if no receiver responds, the router removes that interface from the group and stops sending that group's multicast packets to the interface. Range is 100 to 65,535 ms. Use the **no** form of this command to return to the default setting. |
| **querier-timeout** <*seconds*> | Specifies the interval (in seconds) that the router waits after the current querier's last query before it takes over as querier (IGMP V2). Range is 60 to 300 seconds. Use the **no** form of this command to return to the default setting. |
| **query-interval** <*seconds*> | Specifies the interval (in seconds) at which IGMP queries are sent on an interface. Host query messages are addressed to the all-hosts multicast group with an IP TTL of 1. The router uses queries to detect whether multicast group members are on the interface and to select an IGMP designated router for the attached segment (if more than one multicast router exists). Only the designated router for the segment sends queries. For IGMP V2, the designated router is the router with the lowest IP address on the segment. Range is 0 to 65,535 seconds. Use the **no** form of this command to return to the default setting. |
| **query-max-response-time** <*seconds*> | Specifies the maximum response time (in seconds) advertised by this interface in queries when using IGMP V2. Hosts are allowed a random time within this period to respond, reducing response bursts. Use the **no** form of this command to return to the default setting. |

## Syntax Description

| | |
|---|---|
| **static-group** <*address*> | Configures the router's interface to be a statically-connected member of the specified group. Packets received on the correct RPF interface are forwarded to this interface regardless of whether any receivers have joined the specified group using IGMP. Use the **no** form of this command to remove a configured static group. |
| **version [1 \| 2]** | Sets the interface's IGMP version. Use the **no** form of this command to return to the default setting. |

## Default Values

| | |
|---|---|
| **ip igmp immediate-leave** | No default |
| **ip igmp last-member-query-interval** | 1000 milliseconds |
| **ip igmp querier-timeout** | 2x the **query-interval** value |
| **ip igmp query-interval** | 60 seconds |
| **ip igmp query-max-response-time** | 10 seconds |
| **ip igmp static-group** | No default |
| **ip igmp version** | Version 1 |

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |
| Release 8.1 | ATM sub-interface was added. |
| Release 9.1 | Tunnel sub-interface was added. |

## Usage Examples

The following example sets the query message interval on the interface to 200 milliseconds:

config)**#interface tunnel 1**
(config-tunnel 1)**#ip igmp last-member-query-interval 200**

# ip mcast-stub downstream

Use the **ip mcast-stub downstream** command to enable multicast forwarding and IGMP (router mode) on an interface, and to place it in multicast stub downstream mode. Use the **no** form of this command to disable.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |
| Release 9.1 | Command was expanded to include tunnel interfaces. |

## Functional Notes

This command is used in IP multicast stub applications in conjunction with the **ip mcast-stub helper-address** and **ip mcast-stub upstream** commands. Downstream interfaces connect to segments with multicast hosts. Multiple interfaces may be configured in downstream mode; however, interfaces connecting to the multicast network (upstream) should not be configured in downstream mode. Interfaces configured as downstream should have the lowest IP address of all IGMP-capable routers on the connected segment in order to be selected as the designated router and ensure proper forwarding. Refer to *ip mcast-stub helper-address <ip address>* on page 546 and *ip mcast-stub upstream* on page 1498 for more information.

## Usage Examples

The following example enables multicast forwarding and IGMP on the interface:

```
(config)#interface tunnel 1
(config-tunnel 1)#ip mcast-stub downstream
```

# ip mcast-stub fixed

Use the **ip mcast-stub fixed** command to allow forwarding of multicast traffic on a selected interface after enabling multicast routing. Use the **no** form of this command to disable this mode.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Functional Notes

Multicast routing must be enabled prior to setting **ip mcast-stub fixed** on the selected interface. Also, use the **ip igmp static-group** <*A.B.C.D*> command to receive multicast traffic without host-initiated Internet Group Management Protocol (IGMP) activity on the selected interface. Otherwise, all host-initiated IGMP transactions will enter multicast routes on the router's interface involved with IGMP activities.

### Usage Examples

The following example enables multicast traffic forwarding and IGMP on the interface:

(config)**#interface tunnel 1**
(config-tunnel 1)**#ip mcast-stub fixed**

# ip mcast-stub helper-enable

Use the **ip mcast-stub helper-enable** command to assign the **ip mcast-stub helper-address** as the IGMP proxy. Use the **no** form of this command to disable.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |
| Release 9.1 | Command was expanded to include tunnel interfaces. |

## Functional Notes

This command is used in IP multicast stub applications in conjunction with the **ip mcast-stub helper-address**, **ip mcast-stub upstream**, and **ip mcast-stub downstream** commands. When enabled, the interface becomes a helper forwarding interface. The IGMP host function is dynamically enabled and the interface becomes the active upstream interface, enabling the unit to perform as an IGMP proxy. Refer to *ip mcast-stub helper-address <ip address>* on page 546, *ip mcast-stub downstream* on page 1495, and *ip mcast-stub upstream* on page 1498 for more information.

## Usage Examples

The following example sets the helper address as the IGMP proxy:

(config)**#interface tunnel 1**
(config-tunnel 1)**#ip mcast-stub helper-enable**

# ip mcast-stub upstream

Use the **ip mcast-stub upstream** command to enable multicast forwarding on an interface and place it in multicast stub upstream mode. Use the **no** form of this command to disable.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |
| Release 9.1 | Command was expanded to include tunnel interfaces. |

## Functional Notes

This command is used in IP multicast stub applications in conjunction with the **ip mcast-stub helper-address** and **ip mcast-stub downstream** commands. When enabled, the interface becomes a candidate to be a helper forwarding interface. If chosen as the best path toward the helper address by the router's unicast route table, the IGMP host function is dynamically enabled and the interface becomes the active upstream interface, enabling the router to perform as an IGMP proxy. Though multiple interfaces may be candidates, no more than one interface will actively serve as the helper forwarding interface. Refer to *ip mcast-stub helper-address <ip address>* on page 546 and *ip mcast-stub downstream* on page 1495 for more information.

## Usage Examples

The following example enables multicast forwarding on the interface:

(config)**#interface tunnel 1**
(config-tunnel 1)**#ip mcast-stub upstream**

# ip ospf

Use the **ip ospf** command to customize OSPF settings (if needed). Use the **no** form of these commands to return to the default settings. Variations of this command include:

**ip ospf authentication-key** <*password*>
**ip ospf cost** <*value*>
**ip ospf dead-interval** <*seconds*>
**ip ospf hello-interval** <*seconds*>
**ip ospf message-digest-key [1 | 2] md5** <*key*>
**ip ospf priority** <*value*>
**ip ospf retransmit-interval** <*seconds*>
**ip ospf transmit-delay** <*seconds*>

## Syntax Description

| | |
|---|---|
| **authentication-key** <*password*> | Assigns a simple-text authentication password to be used by other routers using the OSPF simple password authentication. |
| **cost** <*value*> | Specifies the OSPF cost of sending a packet on the interface. This value overrides any computed cost value. Range is 1 to 65,535. |
| **dead-interval** <*seconds*> | Sets the maximum interval (in seconds) allowed between hello packets. If the maximum is exceeded, neighboring devices will determine that the device is down. Range is 0 to 32,767 seconds. |
| **hello-interval** <*seconds*> | Specifies the interval (in seconds) between hello packets sent on the interface. Range is 0 to 32,767 seconds. |
| **message-digest-key [1 \| 2] md5** <*key*> | Configures OSPF Message Digest 5 (MD5) authentication (16-byte max) keys. |
| **priority** <*value*> | Sets the OSPF priority. The value set in this field helps determine the designated router for this network. Range is 0 to 255. |
| **retransmit-interval** <*seconds*> | Specifies the interval (in seconds) between link-state advertisements (LSAs). Range is 0 to 32,767 seconds. |
| **transmit-delay** <*seconds*> | Sets the estimated time (in seconds) required to send an LSA on the interface. Range is 0 to 32,767 seconds. |

## Default Values

| | |
|---|---|
| **dead-interval** <*seconds*> | 40 seconds |
| **hello-interval** <*seconds*> | 10 seconds: Ethernet, point-to-point, Frame Relay, Tunnel, and PPP |
| **retransmit-interval** <*seconds*> | 5 seconds |
| **transmit-delay** <*seconds*> | 1 second |

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |
| Release 9.1 | Command was expanded to include tunnel interfaces. |

## Usage Examples

The following example sets the maximum number of seconds allowed between hello packets to 25,000:

(config)**#interface tunnel 1**
(config-tunnel 1)**#ip ospf dead-interval 25000**

# ip ospf authentication

Use the **ip ospf authentication** command to authenticate an interface that is performing OSPF authentication. Use the **no** form of this command to return to the default setting. Variations of this command include:

**ip ospf authentication**
**ip ospf authentication message-digest**
**ip ospf authentication null**

## Syntax Description

| | |
|---|---|
| **message-digest** | Optional. Selects message-digest authentication type. |
| **null** | Optional. Specifies that no authentication is used. |

## Default Values

By default, this is set to null (meaning no authentication is used).

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |
| Release 9.1 | Command was expanded to include tunnel interfaces. |

## Usage Examples

The following example specifies that no authentication will be used on the tunnel interface:

(config)**#interface tunnel 1**
(config-tunnel 1)**#ip ospf authentication null**

# ip ospf network

Use the **ip ospf network** command to specify the type of network on this interface. Use the **no** form of this command to return to the default setting. Variations of this command include:

**ip ospf network broadcast**
**ip ospf network point-to-point**

## Syntax Description

| | |
|---|---|
| **broadcast** | Sets the network type for broadcast. |
| **point-to-point** | Sets the network type for point-to-point. |

## Default Values

By default, Ethernet defaults to broadcast. PPP, Frame Relay, and tunnel default to point-to-point.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |
| Release 9.1 | Command was expanded to include tunnel interfaces. |

## Functional Notes

A point-to-point network will not elect designated routers.

## Usage Examples

The following example designates a broadcast network type:

(config)**#interface tunnel 1**
(config-tunnel 1)**#ip ospf network broadcast**

# ip pim sparse-mode

Use the **ip pim sparse-mode** command to enable protocol-independent multicast (PIM) sparse mode for this interface. Use the **no** form of this command to disable PIM sparse mode.

## Syntax Description

No subcommands.

## Default Values

By default, PIM sparse mode for this interface is disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Functional Notes

PIM Sparse Mode is a multicast routing protocol that makes use of the unicast forwarding table. It builds unidirectional shared trees rooted at a Rendezvous Point (RP) for a multicast group or a shortest-path tree rooted at a specific source for a multicast group.

## Usage Examples

The following example enables PIM sparse mode on the interface:

(config)**#interface tunnel 1**
(config-tunnel 1)**#ip pim sparse-mode**

# ip pim-sparse dr-priority <*value*>

Use the **ip pim-sparse dr-priority** command to specify the priority for the designated router (DR). This command modifies the routers priority in the DR election process. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the priority of this interface (to be used when determining the DR). Valid range is 1 to 4,294,967,295. |

## Default Values

By default, the priority of all PIM interfaces is 1.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Functional Notes

Interfaces advertise their configured priority values in the hello messages transmitted on the interface. Routers use the priority values to determine the appropriate DR. The router on the network segment with the highest priority is selected as the DR. If a hello message is received on the interface from a router on the network segment and it does not contain a priority, the entire network segment defaults to DR selection based on IP addresses instead of priority. In this instance, the DR is selected as the router on the network segment that has the highest IP address. AOS will always include a priority in all transmitted hello messages. If no priority is specifically designated by the user, the priority is set as the default of 1.

## Usage Examples

The following example specifies a priority of **100** on the Tunnel 1 interface:

(config)**#interface tunnel 1**
(config-tunnel 1)**#ip pim-sparse dr-priority 100**

# ip pim-sparse hello-timer <*value*>

Use the **ip pim-sparse hello-timer** command to specify protocol-independent multicast (PIM) sparse hello timer period. This is the time interval at which periodic hellos are sent out on all interfaces of a PIM-capable router. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the interval (in seconds) at which periodic hellos are sent out of the interface. Valid range is 10 to 3600 seconds. |

## Default Values

By default, the hellos are transmitted on PIM interfaces every 60 seconds.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Functional Notes

Hello messages are used to inform neighbors of a router's presence. Hello messages normally generate a small amount of traffic on an interface. Setting the **hello-timer** to a small interval increases the number of hellos sent (thus increasing the amount of traffic). Set the **hello-timer** to a reasonable value, taking into consideration the bandwidth available on the interface.

## Usage Examples

The following example specifies hellos be sent on the Tunnel 1 interface every **3600** seconds:

(config)**#interface tunnel 1**
(config-tunnel 1)**#ip pim-sparse hello-timer 3600**

# ip pim-sparse nbr-timeout <*value*>

Use the **ip pim-sparse nbr-timeout** command to specify protocol-independent multicast (PIM) sparse neighbor timeout. This is the time interval after which a PIM-capable router will consider a neighbor not present. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the time interval in seconds after which a neighbor is considered not present. Valid range is 30 to 10,800 seconds. |

## Default Values

By default, the nbr-timeout is set to 105 seconds.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example sets the nbr-timeout to **300** seconds:

(config)**#interface tunnel 1**
(config-tunnel 1)**#ip pim-sparse nbr-timeout 300**

# ip pim-sparse override-interval <*value*>

Use the **ip pim-sparse override-interval** command to specify the protocol-independent multicast (PIM) sparse join/prune override interval. This delay interval is the period after a join/prune that another router on the LAN may override a join/prune. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the delay time in milliseconds. Valid range is 0 to 65,535 milliseconds. |

## Default Values

By default, the override interval is set to 2500 milliseconds.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example sets the override interval to **3000** milliseconds:

(config)**#interface tunnel 1**
(config-tunnel 1)**#ip pim-sparse override-interval 3000**

# ip pim-sparse propagation-delay <*value*>

Use the **ip pim-sparse propagation-delay** command to specify the expected propagation delay for join/prune messages. Set the propagation delay (in milliseconds) to estimate the amount of delay found in the local link. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the expected propagation delay in the local link in milliseconds. Valid range is 0 to 32,767 milliseconds. |

## Default Values

By default, the propagation delay is set to 500 milliseconds.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example sets the propagation delay to **300** milliseconds:

(config)**#interface tunnel 1**
(config-tunnel 1)**#ip pim-sparse propagation-delay 300**

# ip policy route-map *<name>*

Use the **ip policy route-map** command to assign a policy route-map to this interface. Use the **no** form of this command to remove the route-map policy.

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies the name of the policy route map to assign to this interface. |

## Default Values

By default, no policy route map is assigned to this interface.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example assigns the policy route map **policy1** to the interface:

(config)**#interface tunnel 1**
(config-tunnel 1)**#ip policy route-map policy1**

# ip proxy-arp

Use the **ip proxy-arp** command to enable proxy Address Resolution Protocol (ARP) on the interface. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, proxy ARP is enabled.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Functional Notes

In general, the principle of proxy ARP allows a router to insert its IP address in the source IP address field of a packet (if the packet is from a host on one of its subnetworks). This allows hosts to reach devices on other subnetworks without implementing routing or specifying a default gateway.

If proxy ARP is enabled, the AOS will respond to all proxy ARP requests with its specified MAC address and forward packets accordingly.

Enabling proxy ARP on an interface may introduce unnecessary ARP traffic on the network.

## Usage Examples

The following enables proxy ARP on the tunnel interface:

(config)**#interface tunnel 1**
(config-tunnel 1)**#ip proxy-arp**

# ip rip receive version

Use the **ip rip receive version** command to configure the RIP version the unit accepts in all RIP packets received on the interface. Use the **no** form of this command to restore the default value. Variations of this command include:

**ip rip receive version 1**
**ip rip receive version 2**

## Syntax Description

| | |
|---|---|
| **1** | Only accept received RIP version 1 packets on the interface. |
| **2** | Only accept received RIP version 2 packets on the interface. |

## Default Values

By default, all interfaces implement RIP version 1 (the default value for the version command).

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Functional Notes

Use the **ip rip receive version** command to specify a RIP version that overrides the **version** (in the Router RIP) configuration. Refer to *version* on page 1734 for more information.

The AOS only accepts one version (either 1 or 2) on a given interface.

## Usage Examples

The following example configures the tunnel interface to accept only RIP version 2 packets:

(config)**#interface tunnel 1**
(config-tunnel 1)**#ip rip receive version 2**

# ip rip send version

Use the **ip rip send version** command to configure the RIP version the unit sends in all RIP packets transmitted on the interface. Use the **no** form of this command to restore the default value. Variations of this command include:

**ip rip send version 1**
**ip rip send version 2**

## Syntax Description

| | |
|---|---|
| **1** | Only transmits RIP version 1 packets on the interface. |
| **2** | Only transmits RIP version 2 packets on the interface. |

## Default Values

By default, all interfaces transmit RIP version 1 (the default value for the **version** command).

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Functional Notes

Use the **ip rip send version** command to specify a RIP version that overrides the **version** (in the Router RIP) configuration. Refer to *version* on page 1734 for more information.

The AOS only transmits one version (either 1 or 2) on a given interface.

## Usage Examples

The following example configures the tunnel interface to transmit only RIP version 2 packets:

(config)**#interface tunnel 1**
(config-tunnel 1)**#ip rip send version 2**

# ip rip summary-address *<ip address> <subnet mask>*

Use the **ip rip summary-address** command to manually summarize the routes Routing Information Protocol (RIP) will advertise and send out a specified interface. Use the **no** form of this command to disable this mode.

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies the summarized network IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.0**). |
| *<subnet mask>* | Specifies the subnet mask that corresponds to the range of IP addresses (network). Subnet masks can be expressed in dotted decimal notation (for example, **255.255.255.0**) or as a prefix length (for example, **/24**). |

## Default Values

By default, no manual summarization is applied by RIP.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Functional Notes

Unlike the automatic summarization on classful network boundaries, only specific network advertisements are made by RIP using the **ip rip summary-address** command. This command is only effective if RIP version 2 is configured.

## Usage Examples

The following example enables manual summarization on the specified IP address:

(config)**#interface tunnel 1**
(config-tunnel 1)**#ip rip summary-address 10.10.123.0 255.255.255.0**

# ip route-cache

Use the **ip route-cache** command to enable fast-cache switching on the interface. Use the **no** form of this command to disable fast-cache switching and return to process switching mode.

*Using Network Address Translation (NAT) or the AOS firewall capabilities on an interface requires process switching mode (using the **no ip route-cache** command).*

## Syntax Description

No subcommands.

## Default Values

By default, fast-cache switching is enabled on all Ethernet and virtual Frame Relay sub-interfaces. IP route-cache is enabled for all virtual PPP interfaces.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Functional Notes

Fast switching allows an IP interface to provide optimum performance when processing IP traffic.

## Usage Examples

The following example enables fast switching on the tunnel interface:

(config)**#interface tunnel 1**
(config-tunnel 1)**#ip route-cache**

# ip urlfilter *<name>*

Use the **ip urlfilter** command to apply a URL filter to the interface for all inbound or outbound traffic. Use the **no** form of this command to remove the URL filter from an interface. Variations of this command include:

**ip urlfilter** *<name>* **in**
**ip urlfilter** *<name>* **out**

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies the URL filter name to use on the interface. |
| **in** | Applies the filter to the inbound traffic. |
| **out** | Applies the filter to the outbound traffic. |

## Default Values

By default, there are no URL filters applied to any interfaces.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Functional Notes

The firewall must be enabled using the **ip firewall** command in order to use URL filters. The URL filter must be created by using the **ip urlfilter *<filtername>* http** command before applying it to the interface. See *ip urlfilter <name> http* on page 598 for more information on using this command.

## Usage Example

The following example performs URL filtering on all traffic entering through the tunnel interface and matches the URL filter named **MyFilter**:

(config)**#interface tunnel 1**
(config-tunnel 1)**#ip urlfilter MyFilter in**

# keepalive

Use the **keepalive** command to periodically send keepalive packets to verify the integrity of the tunnel from end to end. Use the **no** form of this command to disable keepalives. Variations of this command include:

**keepalive**
**keepalive** *<value>*
**keepalive** *<value> <number>*

## Syntax Description

| | |
|---|---|
| *<value>* | Defines the time interval (in seconds) between transmitted keepalive packets. Valid range is 1 to 32,767 seconds. |
| *<number>* | Defines the number of times to retry after failed keepalives before determining that the tunnel endpoint is down. Valid range is 1 to 255 times. |

## Default Values

By default, keepalives are disabled. When enabled, the keepalive period defaults to 10 seconds and the retry count defaults to 3 times.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Functional Notes

Keepalives do not have to be configured on both ends of the tunnel in order to work. A tunnel is not aware of incoming keepalive packets.

## Usage Examples

The following example enables **keepalive** with a period of 30 seconds and a retry count of 5 times:

(config)**#interface tunnel 1**
(config-tunnel 1)**#keepalive 30 5**

# lldp receive

Use the **lldp receive** command to allow LLDP packets to be received on this interface. Use the **no** form of this command to return to the default setting.

## Syntax Description

No subcommands.

## Default Values

By default, all interfaces are configured to send and receive LLDP packets.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Usage Examples

The following example configures the tunnel interface to receive LLDP packets:

(config)**#interface tunnel 1**
(config-tunnel 1)**#lldp receive**

# lldp send

Use the **lldp send** command to configure this interface to transmit LLDP packets or to control the types of information contained in the LLDP packets transmitted by this interface. Use the **no** form of these commands to disable these features. Variations of this command include:

**lldp send**
**lldp send management-address**
**lldp send port-description**
**lldp send system-capabilities**
**lldp send system-description**
**lldp send system-name**
**lldp send-and-receive**

## Syntax Description

| | |
|---|---|
| **management-address** | Enables transmission of management address information on this interface. |
| **port-description** | Enables transmission of port description information on this interface. |
| **system-capabilities** | Enables transmission of this device's system capabilities on this interface. |
| **system-description** | Enables transmission of this device's system description on this interface. |
| **system-name** | Enables transmission of this device's system name on this interface. |
| **and-receive** | Configures this interface to both transmit and receive LLDP packets. |

## Default Values

Be default, all interfaces are configured to transmit and receive LLDP packets of all types.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Functional Notes

Individual LLDP information can be enabled or disabled using the various forms of the **lldp send** command. For example, use the **lldp send and-receive** command to enable transmit and receive of all LLDP information. Then use the **no lldp send port-description** command to prevent LLDP from transmitting port description information.

## Usage Examples

The following example configures the tunnel interface to transmit LLDP packets containing all enabled information types:

(config)**#interface tunnel 1**
(config-tunnel 1)**#lldp send**

The following example configures the tunnel interface to transmit and receive LLDP packets containing all information types:

(config)**#interface tunnel 1**
(config-tunnel 1)**#lldp send and-receive**

# media-gateway ip

Use the **media-gateway ip** command to associate an IP address source to use for RTP traffic. When configuring VoIP, RTP traffic needs an IP address to be associated with it. However, some interfaces allow "dynamic" configuration of IP addresses, and thus, this value could change periodically. Use the **no** form of these commands to disable these functions. Variations of this command include:

**media-gateway ip loopback** <*ip address*>
**media-gateway ip primary**
**media-gateway ip secondary** <*ip address*>

## Syntax Description

| | |
|---|---|
| **loopback** <*ip address*> | Use an IP address statically defined to a loopback interface. Helpful when using a single IP address across multiple WAN interfaces for RTP traffic. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| **primary** | Use the IP address that is configured as primary on this interface. Applies to static, DHCP, or negotiated addresses. |
| **secondary** <*ip address*> | Use the statically defined secondary IP address of this interface to be used for RTP traffic. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

## Default Values

By default, **media-gateway ip** is disabled.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example configures the unit to use the primary IP address for RTP traffic:

(config)**#interface tunnel 1**
(config-tunnel 1)**#media-gateway ip primary**

# mtu *<size>*

Use the **mtu** command to configure the maximum transmit unit (MTU) size for the active interface. Use the **no** form of this command to return to the default value.

## Syntax Description

| | | |
|---|---|---|
| *<size>* | Configures the window size for transmitted packets. The valid ranges for the various interfaces are listed below: | |
| | ATM interfaces | 64 to 1520 |
| | Demand interfaces | 64 to 1520 |
| | Ethernet interfaces | 64 to 1500 |
| | FDL interfaces | 64 to 256 |
| | HDLC interfaces | 64 to 1520 |
| | Loopback interfaces | 64 to 1500 |
| | Tunnel interfaces | 64 to 18,190 |
| | Virtual Frame Relay sub-interfaces | 64 to 1520 |
| | Virtual PPP interfaces | 64 to 1500 |

## Default Values

| | | |
|---|---|---|
| *<size>* | The default values for the various interfaces are listed below: | |
| | ATM interfaces | 1500 |
| | Demand interfaces | 1500 |
| | Ethernet interfaces | 1500 |
| | FDL interfaces | 256 |
| | HDLC interfaces | 1500 |
| | Loopback interfaces | 1500 |
| | Tunnel interfaces | 1500 |
| | Virtual Frame Relay sub-interfaces | 1500 |
| | Virtual PPP interfaces | 1500 |

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

OSPF will not become adjacent on links where the MTU sizes do not match. If router A and router B are exchanging hello packets but their MTU sizes do not match, they will never reach adjacency. This is by design and required by the RFC.

## Usage Examples

The following example specifies an MTU of 1200 on the tunnel interface:

(config)**#interface tunnel 1**
(config-tunnel 1)**#mtu 1200**

# tunnel checksum

Use the **tunnel checksum** command to verify the checksum of incoming Generic Routing Encapsulation (GRE) packets and to include a checksum on outgoing packets. Use the **no** form of this command to disable checksum.

## Syntax Description

No subcommands.

## Default Values

By default, **tunnel checksum** is disabled.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Functional Notes

Both ends of the tunnel must have **tunnel checksum** enabled in order for a meaningful configuration. When both endpoints have **tunnel checksum** enabled, a packet with an incorrect checksum will be dropped. If the endpoints differ in their checksum configuration, all packets will still flow without any checksum verification.

## Usage Examples

The following example enables checksum on the tunnel 1 interface:

(config)**#interface tunnel 1**
(config-tunnel 1)**#tunnel checksum**

## Technology Review

When enabled, the **tunnel checksum** will be calculated for each outgoing GRE packet with the result stored in the GRE header. The checksum present bit will also be set in the header.

# tunnel destination *<ip address>*

Use the **tunnel destination** command to specify the IP address to use as the destination address for all packets transmitted on this interface. Use the **no** form of this command to clear the **tunnel destination** address.

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies the IP address to use as the destination address for all packets transmitted on this interface. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

## Default Values

By default, no tunnel destinations are defined.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Functional Notes

Until a tunnel interface has a destination IP address defined, it is not operational.

The tunnel destination IP address will be the value put into the destination field of the outer IP header after GRE encapsulation of the original packet. A route must be defined for the destination address. Be certain there are no recursive routes by ensuring that a tunnel's destination address will be routed out a physical interface. There is a possibility of creating a routing loop when tunnel interface traffic gets routed back to the same tunnel interface or to another tunnel interface, which in turn, does not have a route out a physical interface. In either case, the tunnel will go down for a period of one minute, after which it will come back up to determine if the recursive routes have been resolved. This allows time for routing protocols to converge on a valid route. If a static route has caused the recursive routing loop, the tunnel status may oscillate until the route is changed.

## Usage Examples

The following example sets the tunnel destination IP address to **192.22.73.101**:

(config)**#interface tunnel 1**
(config-tunnel 1)**#tunnel destination 192.22.73.101**

# tunnel key *<value>*

Use the **tunnel key** command to specify a value shared by both endpoints of the tunnel that will provide minimal security and delineate between tunnels with the same source and destination addresses. Use the **no** form of this command to disable the key.

## Syntax Description

| | |
|---|---|
| *<value>* | Defines the key value for this tunnel. Valid range is 1 to 4,294,967,294. |

## Default Values

By default, a key is not configured.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Functional Notes

When enabled, the key will be stored in the GRE header and the key present bit will be set. If tunnel keys are used, a matching key value must be defined on both endpoints of the tunnel or packets will be discarded.

## Usage Examples

The following example sets the key on a tunnel interface to a value of **1234**:

(config)**#interface tunnel 1**
(config-tunnel 1)**#tunnel key 1234**

# tunnel mode gre

Use the **tunnel mode gre** command to encapsulate traffic destined for the tunnel interface in a Generic Routing Encapsulation (GRE) header. Use the **no** form of this command to set the tunnel to its default mode.

## Syntax Description

No subcommands.

## Default Values

By default, the tunnel interface will be configured for GRE mode.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Functional Notes

GRE is currently the only allowed mode for tunnel interface operation.

## Usage Examples

The following example configures the tunnel interface for GRE mode:

(config)**#interface tunnel 1**
(config-tunnel 1)**#tunnel mode gre**

# tunnel sequence-datagrams

Use the **tunnel sequence-datagrams** command to enable sequence number checking on incoming Generic Routing Encapsulation (GRE) packets, to drop packets arriving out of order, and to include a sequence number in outgoing packets. Use the **no** form of this command to disable sequence number checking.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Functional Notes

Both ends of the tunnel must have sequence numbering enabled. When both endpoints have sequence numbering enabled, a packet arriving with a sequence number less than the current expected value will be dropped. If the endpoints differ in their sequence numbering configuration, all packets will still flow without any sequence number verification. Be careful enabling sequence number verification on a tunnel. The tunnel can easily become out of sequence due to network conditions outside of the tunnel endpoints. It may be difficult to establish a successful traffic flow after an out of sequence condition occurs.

## Usage Examples

The following example enables sequence number processing on the tunnel interface:

(config)**#interface tunnel 1**
(config-tunnel 1)**#tunnel sequence-datagrams**

## Technology Review

When enabled, the next valid sequence number will be placed in the GRE header of each outgoing packet, and the sequence number present bit will be set.

# tunnel source

Use the **tunnel source** command to specify the IP address or name of a physical interface to use as the source address for all packets transmitted on this interface. Use the **no** form of this command to clear the tunnel source address. Variations of this command include:

**tunnel source** *<ip address>*
**tunnel source** *<ip address> <interface>*

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies the IP address in dotted decimal notation to use as the source address for all packets transmitted on this interface. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| *<interface>* | Specifies the interface that contains the IP address to use as the source address for all packets transmitted on this interface. Specify an interface in the format *<interface type [slot/port \| slot/port.sub-interface id \| interface id \| interface id.sub-interface id]>*. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **tunnel source ?** for a complete list of valid interfaces. |

## Default Values

By default, a tunnel source is not defined.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |
| Release 11.1 | Command was expanded to include demand interfaces. |

## Functional Notes

Until a tunnel interface has a source IP address defined and the physical interface used as the source is operational, the tunnel is not operational.

The tunnel source IP address will be the value put into the source field of the outer IP header after GRE encapsulation of the original packet.

## Usage Examples

The following example sets the tunnel source IP address to **192.22.73.101**:

(config)**#interface tunnel 1**
(config-tunnel 1)**#tunnel source 192.22.73.101**

The following example sets the tunnel source IP address to the address of the Ethernet interface labeled 0/1:

(config)**#interface tunnel 1**
(config-tunnel 1)**#tunnel source eth 0/1**

# VLAN CONFIGURATION COMMAND SET

To activate the VLAN Configuration mode, enter the **vlan** command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#vlan 1
(config-vlan 1)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:



All other commands for this command set are described in this section in alphabetical order.



# media ethernet

Use the **media ethernet** command to set the virtual local area network (VLAN) media type to Ethernet. The only media type currently supported is ethernet. Use the **no** form of this command to reset to default.

## Syntax Description

No subcommands.

## Default Values

By default, media is set to ethernet.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following example sets the media type to Ethernet for VLAN 2:

```
(config)#vlan 2
(config-vlan 2)#media ethernet
```

## **name** <*name*>

Use the **name** command to assign a name to the VLAN. Use the **no** form of this command to remove a name given to a VLAN.

### Syntax Description

| | |
|---|---|
| *<name>* | Assigns a name to the VLAN using 1 to 32 characters. |

### Default Values

By default, the name is set to VLANxxxx, where xxxx represents four numeric digits (including leading zeroes) equal to the VLAN ID number.

### Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

### Functional Notes

The name is limited to 32 characters and must be unique throughout.

### Usage Examples

The following example sets the name of VLAN 2 to **Accounting**:

(config)**#vlan 2**
(config-vlan 2)**#name Accounting**

# state

Use the **state** command to change the state of the VLAN. Variations of this command include:

**state active**
**state suspend**

## Syntax Description

| | |
|---|---|
| **active** | Changes the VLAN state to active. |
| **suspend** | Changes the VLAN state to suspended. |

## Default Values

The default setting is active (once the VLAN has been created).

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following example sets the VLAN **state** to suspended:

(config)**#vlan 2**
(config-vlan 1)**#state suspend**

# VLAN DATABASE CONFIGURATION COMMAND SET

To activate the Config VLAN Database mode, enter the **vlan database** command at the Enable security mode prompt. For example:

```
>enable
#vlan database
(vlan)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:



All other commands for this command set are described in this section in alphabetical order.



# abort

Use the **abort** command to exit the VLAN Database without saving any changes made.

## Syntax Description

No subcommands.

## Default Values

No defaults necessary for this setting.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

The **abort** command discards all configuration changes made since you entered the VLAN Database Configuration (or since the last time you issued the **apply** command). The system then exits out of this mode, returning to the enable (**#**) command prompt. Refer to the command *apply* on page 1535 for more information.

## Usage Examples

The following example exits the VLAN Database without saving the changes made:

```
(config)#vlan database
(vlan)#abort
Discarding all changes and exiting.
#
```

# apply

Use the **apply** command to apply changes without exiting the VLAN Database.

## Syntax Description

No subcommands.

## Default Values

No defaults necessary for this setting.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

Applies changes to the VLAN Database configuration in the running configuration.

## Usage Examples

The following example applies changes made, remaining in the VLAN Database:

```
(config)#vlan database
(vlan)#apply
Changes applied.
(vlan)#
```

# reset

Use the **reset** command to discard all changes made and revert to the previous configuration. The prompt remains in the VLAN Database.

## Syntax Description

No subcommands.

## Default Values

No defaults necessary for this setting.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

The **reset** command discards all changes to the VLAN configuration. The configuration remains the same as it was prior to entering the VLAN Database Configuration (or since the last time you issued the **apply** command). The VLAN Database reverts to the same state it had upon entry. Refer to the command *apply* on page 1535 for more information.

## Usage Examples

The following example resets the unit to the previous configuration (i.e., the last configuration saved using the **apply** or the **exit** command):

```
(config)#vlan database
(vlan)#reset
VLAN configuration has been reset.
(vlan)#
```

## show

Use the **show** command to display different aspects of the VLAN configuration. Variations of this command include:

**show changes**
**show changes** <*vlan id*>
**show current**
**show current** <*vlan id*>
**show proposed**
**show proposed** <*vlan id*>

### Syntax Description

| | |
|---|---|
| <*vlan id*> | Specifies a VLAN ID to display only information for a specific VLAN. Valid VLAN interface ID range is from 1 to 4094. |
| **changes** | Displays the proposed changes to the VLAN configuration. |
| **current** | Displays the current VLAN configuration. |
| **proposed** | Displays the proposed VLAN Database. The proposed version is not part of the running configuration until it is applied (using the **apply** command or the **exit** command). |

### Default Values

No defaults necessary.

### Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

### Usage Examples

The following example shows the proposed VLAN Database configuration which will take effect if an **apply** or **exit** command is issued:

(config)**#vlan database**
(vlan)**#show proposed**

# **vlan** *<vlan id>*

Use the **vlan** command to create a VLAN within the VLAN database. Use the **no** form of this command to delete a previously-created VLAN from the database.

## Syntax Description

| | |
|---|---|
| *<vlan id>* | Specifies a valid VLAN interface ID (1 to 4094). |

## Default Values

No defaults necessary for this setting.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following example creates VLAN 2 only within the VLAN Database. This VLAN is not added to the running configuration until an **exit** or **apply** command is issued:

```
(vlan)#vlan 2
VLAN 2 created.
Name = VLAN0002
(vlan)#
```

The following example removes VLAN 2 from the VLAN Database. This VLAN is not removed from the running configuration until an **exit** or **apply** command is issued:

```
(config)#vlan database
(vlan)#no vlan 2
```

# vlan <*vlan id*> media ethernet

Use the **vlan media ethernet** command to set the VLAN media type to Ethernet. Use the **no** form of this command to reset to the default.

## Syntax Description

| | |
|---|---|
| *<vlan id>* | Specifies a valid VLAN interface ID. Valid range is 1 to 4094. |

## Default Values

By default, **vlan media** is set to ethernet.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following example sets the media type of VLAN 2 to ethernet:

(config)**#vlan database**
(vlan)**#vlan 2 media ethernet**

# **vlan** <*vlan id*> **name** <*name*>

Use the **vlan name** command to assign a name to the VLAN. Use the **no** form of this command to remove an assigned name.

## Syntax Description

| | |
|---|---|
| <*vlan id*> | Specifies a valid VLAN interface ID. Valid range is 1 to 4094. |
| <*name*> | Assigns a name to the VLAN using 1 to 32 characters. |

## Default Values

By default, the assigned name is VLANxxxx; where xxxx represents four numeric digits (including leading zeroes) equal to the VLAN ID number.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

The name is limited to 32 characters and must be unique throughout the network.

## Usage Examples

The following example sets the name of VLAN 2 to **Accounting**:

(config)**#vlan database**
(vlan)**#vlan 2 name Accounting**

# vlan <*vlan id*> state

Use the **vlan state** command to change the state of the VLAN. Use the **no** form of this command to return to the default setting. Variations of this command include:

**vlan** <*vlan id*> **state active**
**vlan** <*vlan id*> **state suspend**

## Syntax Description

| | |
|---|---|
| <*vlan id*> | Specifies a valid VLAN interface ID (1 to 4094). |
| **active** | Changes the VLAN state to active. |
| **suspend** | Changes the VLAN state to suspended. |

## Default Values

The default setting is active (once the VLAN has been created).

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following example sets the VLAN state to suspended:

(config)**#vlan database**
(vlan)**#vlan 2 state suspend**

# VLAN INTERFACE CONFIGURATION COMMAND SET

To activate the VLAN Interface Configuration mode, enter the **interface vlan** command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#interface vlan 1
(config-vlan 1)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:



All other commands for this command set are described in this section in alphabetical order.



# access-policy *<name>*

Use the **access-policy** command to assign a specified access policy to an interface. Access policies are applied to traffic entering an interface. Use the **no** form of this command to remove an access policy association. For more information on using access policies, refer to *ip policy-class <name>* on page 549.

*Configured access policies will only be active if the **ip firewall** command has been entered at the Global Configuration mode prompt to enable the AOS security features. All configuration parameters are valid, but no security data processing will be attempted unless the security features are enabled.*

## Syntax Description

| | |
|---|---|
| *<name>* | Identifies the configured access policy by alphanumeric descriptor (all access policy descriptors are case-sensitive). |

## Default Values

By default, there are no configured access policies associated with an interface.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include Ethernet sub-interfaces. |
| Release 6.1 | Command was expanded to include VLAN interfaces. |

## Functional Notes

To assign an access policy to an interface, enter the interface configuration mode for the desired interface and enter **access-policy** *<policy name>*.

## Usage Examples

The following example associates the access policy **Private** (to allow inbound traffic to the Web server) to the VLAN interface 1:

Enable the AOS security features:
(config)**#ip firewall**

Associate the access policy with the VLAN interface 1:

(config)**#interface vlan 1**
(config-interface-vlan 1)**#access-policy Private**

# arp arpa

Use the **arp arpa** command to set ARPA as the standard address resolution protocol (ARP) on this interface. Use the **no** form of this command to return to the default settings.

## Syntax Description

No subcommands.

## Default Values

By default, the ARP is set to ARPA.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |
| Release 6.1 | Command was extended to include NetVanta 2000 Series units. |

## Usage Examples

The following example enables standard ARP for the VLAN interface:

(config)**#interface vlan 1**
(config-interface-vlan 1)**#arp arpa**

# bandwidth *<value>*

Use the **bandwidth** command to provide the bandwidth value of an interface to the higher-level protocols. This value is used in cost calculations. Use the **no** form of this command to restore the default values.

## Syntax Description

| | |
|---|---|
| *<value>* | Specifies bandwidth in kbps. Range is 1 to 4,294,967,295 kbps. |

## Default Values

To view default values, use the **show interfaces** command.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include Ethernet sub-interfaces. |
| Release 6.1 | Command was expanded to include VLAN interfaces. |

## Functional Notes

The **bandwidth** command is an informational value that is communicated to the higher-level protocols to be used in cost calculations. This is a routing parameter only and does not affect the physical interface.

## Usage Examples

The following example sets bandwidth of the VLAN 1 interface to 10 Mbps:

(config)**#interface vlan 1**
(config-interface-vlan 1)**#bandwidth 10000**

# bridge-group *<number>*

Use the **bridge-group** command to assign an interface to the specified bridge group. Use the **no** form of this command to remove the interface from the bridge group.

## Syntax Description

| | |
|---|---|
| *<number>* | Specifies a bridge group number. Range is 1 to 255. |

## Default Values

By default, there are no configured bridge groups.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

A bridged network can provide excellent traffic management to reduce collisions and limit the amount of bandwidth wasted with unnecessary transmissions when routing is not necessary. Any two interfaces can be bridged (e.g., Ethernet to T1 bridge, Ethernet to Frame Relay sub-interface).

## Usage Examples

The following example assigns the VLAN interface to bridge group 17:

(config)**#interface vlan 1**
(config-interface-vlan 1)**#bridge-group 17**

# crypto map *<name>*

Use the **crypto map** command to associate crypto maps with the interface. Use the **no** form of this command to remove a crypto map from an interface.

> NOTE *When you apply a map to an interface, you are applying all crypto maps with the given map name. This allows you to apply multiple crypto maps if you have created maps which share the same name but have different map index numbers.*

> NOTE *For VPN configuration example scripts, refer to the technical support note **Configuring VPN** located on the ADTRAN OS System Documentation CD provided with your unit.*

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies the crypto map name that you wish to assign to the interface. |

## Default Values

By default, no crypto maps are assigned to an interface.

## Command History

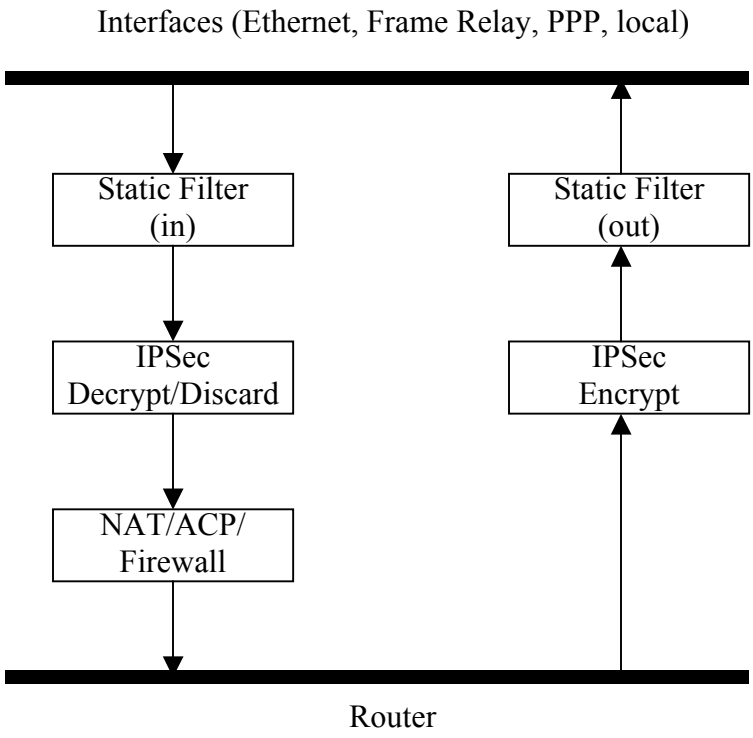
| | |
|---|---|
| Release 4.1 | Command was introduced. |

## Functional Notes

When configuring a system to use both the stateful inspection firewall and IKE negotiation for VPN, keep the following notes in mind.

When defining the policy-class and associated access-control lists (ACLs) that describe the behavior of the firewall, do not forget to include the traffic coming into the system over a VPN tunnel terminated by the system. The firewall should be set up with respect to the unencrypted traffic that is destined to be sent or received over the VPN tunnel. The following diagram represents typical AOS data-flow logic.




As shown in the diagram above, data coming into the product is first processed by the static filter associated with the interface on which the data is received. This access-group is a true static filter and is available for use regardless of whether the firewall is enabled or disabled. Next (if the data is encrypted) it is sent to the IPSec engine for decryption. The decrypted data is then processed by the stateful inspection firewall. Therefore, given a terminating VPN tunnel, only unencrypted data is processed by the firewall.

The ACLs for a crypto map on an interface work in reverse logic to the ACLs for a policy class on an interface. When specifying the ACLs for a crypto map, the source information is the private local side, unencrypted source of the data. The destination information will be the far end, unencrypted destination of the data. However, ACLs for a policy class work in reverse. The source information for the ACL in a policy class is the far end. The destination information is the local side.

## Usage Examples

The following example applies all crypto maps with the name **MyMap** to the VLAN interface:

(config)**#interface vlan 1**
(config-interface-vlan 1)**#crypto map MyMap**

# dynamic-dns

Use the **dynamic-dns** command to configure Dynamic DNS service provided by Dynamic Network Services, Inc. (www.dyndns.org). Use the **no** versions of these commands to disable these features. Variations of this command include:

**dynamic-dns custom** <*hostname*> <*minutes*>
**dynamic-dns dyndns** <*hostname*> <*username*> <*password*>
**dynamic-dns dyndns-custom** <*hostname*> <*username*> <*password*>
**dynamic-dns dyndns-static** <*hostname*> <*username*> <*password*>

## Syntax Description

| | |
|---|---|
| <*hostname*> | Specifies the host name for the server that updates the Dynamic Domain Name Server (DNS). |
| <*minutes*> | Specifies the intervals in minutes to update the server with information (updates also occur when the interface's IP address changes regardless of the update intervals).<br>Refer to *Functional Notes* below for additional argument descriptions. |

## Default Values

No default is necessary for this command.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |
| Release 12.1 | Command was expanded. |

## Functional Notes

**custom** - Constanttime.com's Custom Dynamic DNS$^{SM}$ service allows you complete access and management control over your domain name regardless of where you purchased/registered it. This allows to manage IP address mappings (A records), domain aliases (CNAME records) and mail servers (MX records).

**dyndns** - The Dynamic DNS$^{SM}$ offered by Dynamic Network Services Inc., (DynDNS.org) allows you to alias a dynamic IP address to a static host name in various domains. This allows your unit to be more easily accessed from various locations on the Internet. This service is provided for up to five host names.

**dyndns-custom -** DynDNS.org's Custom DNS$^{SM}$ service provides a full DNS solution, giving you complete control over an entire domain name. A web-based interface provides two levels of control over your domain, catering to average or advanced users. Five globally redundant DNS servers ensure that your domain will always resolve.

A choice of two interfaces is available. The basic interface is designed for most users. It comes preconfigured for the most common configuration and allows for easy creation of most common record types. The advanced interface is designed for system administrators with a solid DNS background, and provides layout and functionality similar to a BIND zone file allowing for the creation of nearly any record type.

Custom DNS$^{SM}$ can be used with both static and dynamic IPs and has the same automatic update capability through Custom DNS-aware clients as Dynamic DNS.

**dyndns-static -** The Static DNS service is similar to DynDNS.org's Dynamic DNS$^{SM}$ service in that it allows a host name such as yourname.dyndns.org to point to your IP address. Unlike a Dynamic DNS host, a Static DNS host does not expire after 35 days without updates, but updates take longer to propagate though the DNS system. This service is provided for up to five host names.

If your IP address does not change often or at all but you still want an easy name to remember it by (without having to purchase your own domain name), Static DNS service is ideal for you.

If you would like to use your own domain name (such as yourname.com), you need Custom DNS service which also provides full dynamic and static IP address support.

**Usage Examples**

The following example sets the dynamic-dns to **dyndns-custom** with host name **host**, user name **user**, and password **pass**:

(config)**#interface vlan 1**
(config-interface-vlan 1)**#dynamic-dns dyndns-custom host user pass**

# ip access-group *<name>*

Use the **ip access-group** command to create an access control list to be used for packets transmitted on or received from the specified interface. Use the **no** form of this command to disable this type of control. Variations of this command include:

**ip access-group** <*name*> **in**
**ip access-group** <*name*> **out**

## Syntax Description

| | |
|---|---|
| *<name>* | Assigns an IP access control list name. |
| **in** | Enables access control on packets received on the specified interface. |
| **out** | Enables access control on packets transmitted on the specified interface. |

## Default Values

By default, these commands are disabled.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Functional Notes

When this command is enabled, the IP destination address of each packet must be validated before being passed through. If the packet is not acceptable per these settings, it is dropped.

## Usage Examples

The following example sets up the router to only allow Telnet traffic (as defined in the user-configured **TelnetOnly** IP access control list) into the VLAN interface:

(config)**#ip access-list extended TelnetOnly**
(config-ext-nacl)**#permit tcp any any eq telnet**
(config-ext-nacl)**#interface vlan 1**
(config-interface-vlan 1)**#ip access-group TelnetOnly in**

# ip address *<ip address> <subnet mask>*

Use the **ip address** command to define an IP address on the specified interface. Use the **no** form of this command to remove a configured IP address. Variations of this command include:

**ip address** *<ip address> <subnet mask>*
**ip address** *<ip address> <subnet mask>* **secondary**

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies a valid IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| *<subnet mask>* | Specifies the subnet mask that corresponds to a range of IP addresses (network) or a specific host. Subnet masks can be expressed in dotted decimal notation (for example, **255.255.255.0**) or as a prefix length (for example, **/24**). |
| **secondary** | Optional. Configures a secondary IP address for the specified interface. |

## Default Values

By default, there are no assigned IP addresses.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following example configures an IP address of **192.22.72.101/30**:

(config)**#interface vlan 1**
(config-interface-vlan 1)**#ip address 192.22.72.101 255.255.255.252**

# ip address dhcp

Use the **ip address dhcp** command to use Dynamic Host Configuration Protocol (DHCP) to obtain an address on the interface. Use the **no** form of this command to remove a configured IP address (using DHCP) and disable DHCP operation on the interface.

**ip address dhcp client-id [**<*interface*> **|** <*identifier*>**] [hostname** <*"string"*>**] [**<*administrative distance*>**]**
**ip address dhcp hostname** <*"string"*> **[no-default-route | no-domain-name | no-nameservers] [**<*administrative distance*>**]**
**ip address dhcp [no-default-route | no-domain-name | no-nameservers] [**<*administrative distance*>**]**

## Syntax Description

| | |
|---|---|
| <*administrative distance*> | Optional. Specifies the administrative distance to use when adding the DHCP gateway into the route table. It is used to determine the best route when multiple routes to the same destination exist. The smaller the administrative distance the more reliable the route. Range is 1 to 255. |
| **client-id** | Optional. Specifies the client identifier used when obtaining an IP address from a DHCP server. |
| <*interface*> | Specifying an interface defines the client identifier as the hexadecimal MAC address of the specified interface (including a hexadecimal number added to the front of the MAC address to identify the media type).<br>For example, specifying the **client-id ethernet 0/1** (where the Ethernet interface has a MAC address of d217.0491.1150) defines the client identifier as **01:d2:17:04:91:11:50** (where 01 defines the media type as Ethernet). Refer to *hardware-address* <*mac address*> on page 1973 for a detailed listing of media types. |
| <*identifier*> | Specifies a custom client-identifier using a text string (that is converted to a hexadecimal equivalent) or 7 to 28 hexadecimal numbers (with colon delimiters).<br>For example, a custom client identifier of **0f:ff:ff:ff:ff:51:04:99:a1** may be entered using the <*identifier*> option. |
| **hostname** <*"string"*> | Optional. Specifies a text string (to override the global router name) to use as the name in the DHCP option 12 field. The string is enclosed in quotation marks and can consist of up to 35 characters. |
| **no-default-route** | Instructs AOS not install the default-route obtained via DHCP. |
| **no-domain-name** | Instructs AOS not install the domain-name obtained via DHCP. |
| **no-nameservers** | Instructs AOS not install the DNS servers obtained via DHCP. |

## Default Values

| | |
|---|---|
| *<administrative distance>* | By default, the administrative distance value is 1. |
| **client-id** | Optional. By default, the client identifier is populated using the following formula:<br>TYPE: INTERFACE SPECIFIC INFO : MAC ADDRESS<br>Where TYPE specifies the media type in the form of one hexadecimal byte (refer to *hardware-address <mac address>* on page 1973 for a detailed listing of media types), and the MAC ADDRESS is the Media Access Control (MAC) address assigned to the first Ethernet interface in the unit in the form of six hexadecimal bytes. (For units with a single Ethernet interface, the MAC ADDRESS assigned to Ethernet 0/1 is used in this field).<br>INTERFACE SPECIFIC INFO is only used for Frame Relay interfaces and can be determined using the following:<br>FR_PORT# : Q.922 ADDRESS<br>Where the FR_PORT# specifies the label assigned to the virtual Frame Relay interface using four hexadecimal bytes. For example, a virtual Frame Relay interface labeled 1 would have a FR_PORT# of 00:00:00:01.<br>The Q.922 ADDRESS field is populated using the following: |

| 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
|---|---|---|---|---|---|---|---|
| DLCI (high order) | | | | | | C/R | EA |
| DLCI (lower) | | | | FECN | BECN | DE | EA |

Where the FECN, BECN, C/R, DE, and high order EA bits are assumed to be 0 and the lower order extended address (EA) bit is set to 1.

The following list provides a few example DLCIs and associated Q.922 address: DLCI (decimal) / Q.922 address (hex)

16 / 0x0401

50 / 0x0C21

60 / 0x0CC1

70 / 0x1061

80 / 0x1401

| | |
|---|---|
| **hostname** <*"string"*> | By default, the host name is the name configured using the Global Configuration **hostname** command. |

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |
| Release 13.1 | Command was expanded to include administrative distance. |

## Functional Notes

DHCP allows interfaces to acquire a dynamically assigned IP address from a configured DHCP server on the network. Many Internet Service Providers (ISPs) require the use of DHCP when connecting to their services. Using DHCP reduces the number of dedicated IP addresses the ISP must obtain. Consult your ISP to determine the proper values for the **client-id** and **hostname** fields.

## Usage Examples

The following example enables DHCP operation on the VLAN interface (labeled 1):

(config)**#interface vlan 1**
(config-interface-vlan 1)**#ip address dhcp**

The following example enables DHCP operation on the VLAN interface (labeled 1) utilizing hostname **adtran** and does not allow obtaining a default route, domain name, or nameservers. It also sets the administrative distance as **5**:

(config)**#interface vlan 1**
(config-vlan 1)**#ip address dhcp hostname “adtran” no-default-route no-domain-name no-nameservers 5**

# ip dhcp

Use the **ip dhcp** command to release or renew the DHCP IP address. This command is only applicable when using DHCP for IP address assignment. Variations of this command include:

**ip dhcp release**
**ip dhcp renew**

## Syntax Description

| | |
|---|---|
| **release** | Releases DHCP IP address. |
| **renew** | Renews DHCP IP address. |

## Default Values

No default values required for this command.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Usage Examples

The following example releases the IP DHCP address for the VLAN interface:

(config)**#interface vlan 1**
(config-interface-vlan 1)**#ip dhcp release**

# ip ffe

Use the **ip ffe** command to enable the FastFlow Engine (FFE) on this interface with the default number of entries. Use the **no** form of these commands to disable the FFE. Variations of this command include:

**ip ffe max-entries** <*entries*>

*Issuing this command will cause all FFE entries on this interface to be cleared.*

## Syntax Description

| | |
|---|---|
| **max-entries** <*entries*> | Specifies the maximum number of entries stored in the flow table. Valid range is from 1 to 8192. |

## Default Values

By default, the FFE is disabled. The default number of **max-entries** is 4096.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Functional Notes

The FFE can be used to help reduce routing overhead and thus reduce overall routing times. Routing times are reduced by the creation of a flow table on the ingress interface. The maximum number of entries that can be stored in the flow table at any one time may be specified by using the max-entries subcommand. When features such as firewall, VPN, and static filters are enabled, routing time reduction may not be realized.

## Usage Examples

The following example enables the FFE and sets the maximum number of entries in the flow table to 50:

(config)**#interface vlan 1**
(config-interface-vlan 1)**#ip ffe max-entries 50**

## Technology Review

The FastFlow system goal is to increase IP packet throughput by moving as much of the packet processing into the engine as possible. Packets are classified into flows based upon the IP protocol (TCP, UDP, ICMP, etc.), the source and destination IP addresses, IP TOS, and the protocol-specific information such as the source and destination port numbers. Flows are defined as the unidirectional representation of a conversation between two IP hosts. Each ingress interface keeps its own flow table, a collection of flow entries.

The first packet in a flow that is forwarded through the unit will build a flow entry. When a flow entry is looked up but no entry is found, a FastFlowBuilder object is allocated and attached to the packet. As the packet passes through the various processing layers, each subsystem will add processing to the FastFlowBuilder. When packet is about to be forwarded out of the egress interface, the FastFlowBuilder will be finalized. That is, the flow entry being built will be checked for completeness and committed to the flow table on the ingress interface. Subsequent flow matches can then bypass the normal processing layers.

# ip mcast-stub fixed

Use the **ip mcast-stub fixed** command to allow forwarding of multicast traffic on a selected interface after enabling multicast routing. Use the **no** form of this command to disable this mode.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Functional Notes

Multicast routing must be enabled prior to setting **ip mcast-stub fixed** on the selected interface. Also, use the **ip igmp static-group** <*A.B.C.D*> command to receive multicast traffic without host-initiated Internet Group Management Protocol (IGMP) activity on the selected interface. Otherwise, all host-initiated IGMP transactions will enter multicast routes on the router's interface involved with IGMP activities.

### Usage Examples

The following example enables multicast traffic forwarding and IGMP on the interface:

(config)**#vlan 2**
(config-vlan 2)**#ip mcast-stub fixed**

# ip helper-address *<ip address>*

Use the **ip helper-address** command to configure the AOS to forward User Datagram Protocol (UDP) broadcast packets received on the interface. Use the **no** form of this command to disable forwarding packets.

> NOTE *The **ip helper-address** command must be used in conjunction with the **ip forward-protocol** command to configure the AOS to forward UDP broadcast packets. Refer to ip forward-protocol udp <value> on page 532 for more information.*

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies the destination IP address for the forwarded UDP packets. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

## Default Values

By default, broadcast UDP packets are not forwarded.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

When used in conjunction with the **ip forward-protocol** command, the **ip helper-address** feature allows you to customize which broadcast packets are forwarded.

To implement the helper address feature, assign helper address(es) (specifying the device that needs to receive the broadcast traffic) to the interface closest to the host that transmits the broadcast packets. When broadcast packets (of the specified type forwarded using the **ip forward-protocol** command) are received on the interface, they will be forwarded to the device that needs the information.

Only packets meeting the following criteria are considered eligible by the **ip helper-address** feature:

1. The packet IP protocol is User Datagram Protocol (UDP).
2. Any UDP port specified using the **ip forward-protocol** command.
3. The media access control (MAC) address of the frame is an all-ones broadcast address (ffff.ffff.ffff).
4. The destination IP address is broadcast defined by all ones (255.255.255.255) or a subnet broadcast (for example, 192.33.4.251 for the 192.33.4.248/30 subnet).

## Usage Examples

The following example forwards all DNS broadcast traffic to the DNS server with IP address 192.33.5.99:

(config)**#ip forward-protocol udp domain**
(config)**#interface vlan 1**
(config-interface-vlan 1)**#ip helper-address 192.33.5.99**

# ip igmp

Use the **ip igmp** command to configure multicasting-related functions for the interface. Variations of this command include:

**ip igmp immediate-leave**
**ip igmp last-member-query-interval** <*milliseconds*>
**ip igmp querier-timeout** <*seconds*>
**ip igmp query-interval** <*seconds*>
**ip igmp query-max-response-time** <*seconds*>
**ip igmp static-group** <*address*>
**ip igmp version [1 | 2]**

## Syntax Description

| | |
|---|---|
| **immediate-leave** | Specifies that if only one host (or IGMP snooping switch) is connected to the interface, when a leave is received, multicast of that group is immediately terminated as opposed to sending a group query and timing out the group if no device responds. Works in conjunction with **ip igmp last-member-query-interval**. Applies to all groups when configured. Use the **no** form of this command to disable the **immediate-leave** feature. |
| **last-member-query-interval** <*milliseconds*> | Controls the timeout (in milliseconds) used to detect whether any group receivers remain on an interface after a receiver leaves a group. If a receiver sends a leave-group message (IGMP Version 2), the router sends a group-specific query on that interface. After twice the time specified by this command plus as much as one second longer, if no receiver responds, the router removes that interface from the group and stops sending that group's multicast packets to the interface. Range is 100 to 65,535 ms. Use the **no** form of this command to return to the default setting. |
| **querier-timeout** <*seconds*> | Specifies the interval (in seconds) that the router waits after the current querier's last query before it takes over as querier (IGMP V2). Range is 60 to 300 seconds. Use the **no** form of this command to return to the default setting. |
| **query-interval** <*seconds*> | Specifies the interval (in seconds) at which IGMP queries are sent on an interface. Host query messages are addressed to the all-hosts multicast group with an IP TTL of 1. The router uses queries to detect whether multicast group members are on the interface and to select an IGMP designated router for the attached segment (if more than one multicast router exists). Only the designated router for the segment sends queries. For IGMP V2, the designated router is the router with the lowest IP address on the segment. Range is 0 to 65,535 seconds. Use the **no** form of this command to return to the default setting. |
| **query-max-response-time** <*seconds*> | Specifies the maximum response time (in seconds) advertised by this interface in queries when using IGMP V2. Hosts are allowed a random time within this period to respond, reducing response bursts. Use the **no** form of this command to return to the default setting. |

## Syntax Description

| | |
|---|---|
| **static-group** <*address*> | Configures the router's interface to be a statically-connected member of the specified group. Packets received on the correct RPF interface are forwarded to this interface regardless of whether any receivers have joined the specified group using IGMP. Use the **no** form of this command to remove a configured static group. |
| **version [1 \| 2]** | Sets the interface's IGMP version. Use the **no** form of this command to return to the default setting. |

## Default Values

| | |
|---|---|
| **ip igmp immediate-leave** | No default |
| **ip igmp last-member-query-interval** | 1000 milliseconds |
| **ip igmp querier-timeout** | 2x the **query-interval** value |
| **ip igmp query-interval** | 60 seconds |
| **ip igmp query-max-response-time** | 10 seconds |
| **ip igmp static-group** | No default |
| **ip igmp version** | Version 1 |

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |

## Usage Examples

The following example sets the query message interval on the interface to 200 milliseconds:

(config)**#interface vlan 1**
(config-interface-vlan 1)**#ip igmp last-member-query-interval 200**

# ip mcast-stub downstream

Use the **ip mcast-stub downstream** command to enable multicast forwarding and IGMP (router mode) on an interface, and to place it in multicast stub downstream mode. Use the **no** form of this command to disable.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |

## Functional Notes

This command is used in IP multicast stub applications in conjunction with the **ip mcast-stub helper-address** and **ip mcast-stub upstream** commands. Downstream interfaces connect to segments with multicast hosts. Multiple interfaces may be configured in downstream mode; however, interfaces connecting to the multicast network (upstream) should not be configured in downstream mode. Interfaces configured as downstream should have the lowest IP address of all IGMP-capable routers on the connected segment in order to be selected as the designated router and ensure proper forwarding. Refer to *ip mcast-stub helper-address <ip address>* on page 546 and *ip mcast-stub upstream* on page 1567 for more information.

## Usage Examples

The following example enables multicast forwarding and IGMP on the interface:

```
(config)#interface vlan 1
(config-interface-vlan 1)#ip mcast-stub downstream
```

# ip mcast-stub fixed

Use the **ip mcast-stub fixed** command to allow forwarding of multicast traffic on a selected interface after enabling multicast routing. Use the **no** form of this command to disable this mode.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Functional Notes

Multicast routing must be enabled prior to setting **ip mcast-stub fixed** on the selected interface. Also, use the **ip igmp static-group** <*A.B.C.D*> command to receive multicast traffic without host-initiated Internet Group Management Protocol (IGMP) activity on the selected interface. Otherwise, all host-initiated IGMP transactions will enter multicast routes on the router's interface involved with IGMP activities.

### Usage Examples

The following example enables multicast traffic forwarding and IGMP on the interface:

(config)**#vlan 2**
(config-vlan 2)**#ip mcast-stub fixed**

# ip mcast-stub helper-enable

Use the **ip mcast-stub helper-enable** command to assign the **ip mcast-stub helper-address** as the IGMP proxy. Use the **no** form of this command to disable.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |
| Release 10.1 | Command was expanded to include VLAN interfaces. |

## Functional Notes

This command is used in IP multicast stub applications in conjunction with the **ip mcast-stub helper-address**, **ip mcast-stub upstream**, and **ip mcast-stub downstream** commands. When enabled, the interface becomes a helper forwarding interface. The IGMP host function is dynamically enabled and the interface becomes the active upstream interface, enabling the unit to perform as an IGMP proxy. Refer to *ip mcast-stub helper-address <ip address>* on page 546, *ip mcast-stub downstream* on page 1564, and *ip mcast-stub upstream* on page 1567 for more information.

## Usage Examples

The following example sets the helper address as the IGMP proxy:

(config)**#interface vlan 1**
(config-vlan 1)**#ip mcast-stub helper-enable**

# ip mcast-stub upstream

Use the **ip mcast-stub upstream** command to enable multicast forwarding on an interface and place it in multicast stub upstream mode. Use the **no** form of this command to disable.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |

## Functional Notes

This command is used in IP multicast stub applications in conjunction with the **ip mcast-stub helper-address** and **ip mcast-stub downstream** commands. When enabled, the interface becomes a candidate to be a helper forwarding interface. If chosen as the best path toward the helper address by the router's unicast route table, the IGMP host function is dynamically enabled and the interface becomes the active upstream interface, enabling the router to perform as an IGMP proxy. Though multiple interfaces may be candidates, no more than one interface will actively serve as the helper forwarding interface. Refer to *ip mcast-stub helper-address <ip address>* on page 546 and *ip mcast-stub downstream* on page 1564 for more information.

## Usage Examples

The following example enables multicast forwarding on the interface:

```
(config)#interface vlan 1
(config-interface-vlan 1)#ip mcast-stub upstream
```

# ip mtu <*value*>

Use the **ip mtu** command to set the maximum transmission unit size (in bytes) for the VLAN interface. To reset to the default setting, use the **no** version of this command.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the MTU size in bytes. Range is 68 to 1,000,000 bytes. |

## Default Values

By default, this is set to 1500 bytes.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following example configures the IP MTU for 2000 bytes:

(config)**#interface vlan 1**
(config-interface-vlan 1)**#ip mtu 2000**

# ip ospf

Use the **ip ospf** command to customize OSPF settings (if needed). Use the **no** form of these commands to return to the default settings. Variations of this command include:

**ip ospf authentication-key** <*password*>
**ip ospf cost** <*value*>
**ip ospf dead-interval** <*seconds*>
**ip ospf hello-interval** <*seconds*>
**ip ospf message-digest-key [1 | 2] md5** <*key*>
**ip ospf priority** <*value*>
**ip ospf retransmit-interval** <*seconds*>
**ip ospf transmit-delay** <*seconds*>

## Syntax Description

| | |
|---|---|
| **authentication-key** <*password*> | Assigns a simple-text authentication password to be used by other routers using the OSPF simple password authentication. |
| **cost** <*value*> | Specifies the OSPF cost of sending a packet on the interface. This value overrides any computed cost value. Range is 1 to 65,535. |
| **dead-interval** <*seconds*> | Sets the maximum interval (in seconds) allowed between hello packets. If the maximum is exceeded, neighboring devices will determine that the device is down. Range is 0 to 32,767 seconds. |
| **hello-interval** <*seconds*> | Specifies the interval (in seconds) between hello packets sent on the interface. Range is 0 to 32,767 seconds. |
| **message-digest-key [1 \| 2] md5** <*key*> | Configures OSPF Message Digest 5 (MD5) authentication (16-byte max) keys. |
| **priority** <*value*> | Sets the OSPF priority. The value set in this field helps determine the designated router for this network. Range is 0 to 255. |
| **retransmit-interval** <*seconds*> | Specifies the interval (in seconds) between link-state advertisements (LSAs). Range is 0 to 32,767 seconds. |
| **transmit-delay** <*seconds*> | Sets the estimated time (in seconds) required to send an LSA on the interface. Range is 0 to 32,767 seconds. |

## Default Values

| | |
|---|---|
| **dead-interval** <*seconds*> | 40 seconds |
| **hello-interval** <*seconds*> | 10 seconds: Ethernet, point-to-point, Frame Relay, and PPP |
| **retransmit-interval** <*seconds*> | 5 seconds |
| **transmit-delay** <*seconds*> | 1 second |

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Usage Examples

The following example sets the maximum number of seconds allowed between hello packets to 25,000:

(config)**#interface vlan 1**
(config-interface-vlan 1)**#ip ospf dead-interval 25000**

# ip ospf authentication

Use the **ip ospf authentication** command to authenticate an interface that is performing OSPF authentication. Use the **no** form of this command to return to the default setting. Variations of this command include:

**ip ospf authentication**
**ip ospf authentication message-digest**
**ip ospf authentication null**

## Syntax Description

| | |
|---|---|
| **message-digest** | Optional. Selects message-digest authentication type. |
| **null** | Optional. Specifies that no authentication is used. |

## Default Values

By default, this is set to null (meaning no authentication is used).

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Usage Examples

The following example specifies that no authentication will be used on the VLAN interface:

(config)**#interface vlan 1**
(config-interface-vlan 1)**#ip ospf authentication null**

# ip ospf network

Use the **ip ospf network** command to specify the type of network on this interface. Use the **no** form of this command to return to the default setting. Variations of this command include:

**ip ospf network broadcast**
**ip ospf network point-to-point**

## Syntax Description

| | |
|---|---|
| **broadcast** | Sets the network type for broadcast. |
| **point-to-point** | Sets the network type for point-to-point. |

## Default Values

By default, Ethernet defaults to broadcast. PPP and Frame Relay default to point-to-point.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Functional Notes

A point-to-point network will not elect designated routers.

## Usage Examples

The following example designates a broadcast network type:

(config)**#interface vlan 1**
(config-interface-vlan 1)**#ip ospf network broadcast**

# ip proxy-arp

Use the **ip proxy-arp** command to enable proxy Address Resolution Protocol (ARP) on the interface. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, proxy ARP is enabled.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

In general, the principle of proxy ARP allows a router to insert its IP address in the source IP address field of a packet (if the packet is from a host on one of its subnetworks). This allows hosts to reach devices on other subnetworks without implementing routing or specifying a default gateway.

If proxy ARP is enabled, the AOS will respond to all proxy ARP requests with its specified MAC address and forward packets accordingly.

Enabling proxy ARP on an interface may introduce unnecessary ARP traffic on the network.

## Usage Examples

The following enables proxy ARP on the VLAN interface:

(config)**#interface vlan 1**
(config-interface-vlan 1)**#ip proxy-arp**

# ip rip receive version

Use the **ip rip receive version** command to configure the RIP version the unit accepts in all RIP packets received on the interface. Use the **no** form of this command to restore the default value. Variations of this command include:

**ip rip receive version 1**
**ip rip receive version 2**

## Syntax Description

| | |
|---|---|
| **1** | Accepts only RIP version 1 packets received on the interface. |
| **2** | Accepts only RIP version 2 packets received on the interface. |

## Default Values

By default, all interfaces implement RIP version 1 (the default value for the version command).

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

Use the **ip rip receive version** command to specify a RIP version that overrides the **version** (in the Router RIP) configuration. Refer to *version* on page 1734 for more information.

The AOS only accepts one version (either 1 or 2) on a given interface.

## Usage Examples

The following example configures the VLAN interface to accept only RIP version 2 packets:

(config)**#interface vlan 1**
(config-interface-vlan 1)**#ip rip receive version 2**

# ip rip send version

Use the **ip rip send version** command to configure the RIP version the unit sends in all RIP packets transmitted on the interface. Use the **no** form of this command to restore the default value. Variations of this command include:

**ip rip send version 1**
**ip rip send version 2**

## Syntax Description

| | |
|---|---|
| **1** | Transmits only RIP version 1 packets on the interface. |
| **2** | Transmits only RIP version 2 packets on the interface. |

## Default Values

By default, all interfaces transmit RIP version 1 (the default value for the **version** command).

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

Use the **ip rip send version** command to specify a RIP version that overrides the **version** (in the Router RIP) configuration. Refer to *version* on page 1734 for more information.

The AOS only transmits one version (either 1 or 2) on a given interface.

## Usage Examples

The following example configures the VLAN interface to transmit only RIP version 2 packets:

(config)**#interface vlan 1**
(config-interface-vlan 1)**#ip rip send version 2**

# ip rip summary-address *<ip address> <subnet mask>*

Use the **ip rip summary-address** command to manually summarize the routes Routing Information Protocol (RIP) will advertise and send out a specified interface. Use the **no** form of this command to disable this mode.

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies the summarized network IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.0**). |
| *<subnet mask>* | Specifies the subnet mask that corresponds to the range of IP addresses (network). Subnet masks can be expressed in dotted decimal notation (for example, **255.255.255.0**) or as a prefix length (for example, **/24**). |

## Default Values

By default, no manual summarization is applied by RIP.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Functional Notes

Unlike the automatic summarization on classful network boundaries, only specific network advertisements are made by RIP using the **ip rip summary-address** command. This command is only effective if RIP version 2 is configured.

## Usage Examples

The following example enables manual summarization on the specified IP address:

(config)**#interface vlan 1**
(config-interface-vlan 1)**#ip rip summary-address 10.10.123.0 255.255.255.0**

# ip route-cache

Use the **ip route-cache** command to enable fast-cache switching on the interface. Use the **no** form of this command to disable fast-cache switching and return to process switching mode.


*Using Network Address Translation (NAT) or the AOS firewall capabilities on an interface requires process switching mode (using the **no ip route-cache** command).*

## Syntax Description

No subcommands.

## Default Values

By default, fast-cache switching is enabled on all Ethernet and virtual Frame Relay sub-interfaces. IP route cache is enabled for all virtual PPP interfaces.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |

## Functional Notes

Fast switching allows an IP interface to provide optimum performance when processing IP traffic.

## Usage Examples

The following example enables fast switching on the VLAN interface:

(config)**#interface vlan 1**
(config-interface-vlan 1)**#ip route-cache**

# ip unnumbered *<interface>*

Use the **ip unnumbered** command to use the IP address assigned to the specified interface for all IP processing on the active interface. Use the **no** form of this command to remove the unnumbered configuration.

## Syntax Description

| | |
|---|---|
| *<interface>* | Specifies the interface that contains the IP address to use as the source address for all packets transmitted on this interface. Specify an interface in the format *<interface type [slot/port \| slot/port.sub-interface id \| interface id \| interface id.sub-interface id]>*. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **show ip unnumbered ?** for a list of valid interfaces. |

## Default Values

By default, all interfaces are configured to use a specified IP address (using the **ip address** command).

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced |
| Release 11.1 | Command was expanded to include demand interfaces. |

## Functional Notes

If **ip unnumbered** is enabled on an interface, all IP traffic from the interface will use a source IP address taken from the specified interface. For example, specifying **ip unnumbered eth 0/1** while in the Frame Relay Sub-Interface Configuration mode configures the Frame Relay sub-interface to use the IP address assigned to the Ethernet interface for all IP processing. In addition, the AOS uses the specified interface information when sending route updates over the unnumbered interface.

## Usage Examples

The following example configures the VLAN interface (labeled **vlan 1**) to use the IP address assigned to the Ethernet interface (**eth 0/1**):

(config)**#interface vlan 1**
(config-interface-vlan 1)**#ip unnumbered eth 0/1**

# mac-address *<mac address>*

Use the **mac-address** command to specify the Media Access Control (MAC) address of the VLAN interface. Only the last three values of the MAC address can be modified. The first three values contain the ADTRAN reserved number (00:0A:C8) by default. Use the **no** form of this command to return to the default MAC address programmed by ADTRAN.

## Syntax Description

| | |
|---|---|
| *<mac address>* | Specifies a valid 48-bit MAC address. MAC addresses should be expressed in following format xx:xx:xx:xx:xx:xx (for example, **00:A0:C8:00:00:01**). |

## Default Values

A unique default MAC address is programmed in each unit shipped by ADTRAN.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following example configures a MAC address of **00:0A:C8:5F:00:D2**:

(config)**#interface vlan 1**
(config-interface-vlan 1)**#mac-address 00:0A:C8:5F:00:D2**

# max-reserved-bandwidth <*value*>

Use the **max-reserved-bandwidth** command to specify the percentage of interface bandwidth reserved for use in user-defined (priority or class-based) queues. The remainder of the interface bandwidth is reserved for system critical traffic and is not available to user-defined queues. Use the **no** form of this command to restore the default values.


*Reserving a portion of the interface bandwidth for system critical traffic is necessary for proper operation. Specifying the entire interface bandwidth for use in user-defined queues can cause undesirable operation.*

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the maximum percentage of bandwidth to reserve for QoS. This setting is configured as a percentage of the total interface speed. Range is 1 to 100 percent. |

## Default Values

By default, **max-reserved-bandwidth** is set to 75 percent, which reserves 25 percent of the interface bandwidth for system critical traffic.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example specifies **85** percent of the bandwidth on the VLAN 1 interface be available for use in user-defined queues:

(config)**#interface vlan 1**
(config-interface-vlan 1)**#max-reserved-bandwidth 85**

# media-gateway ip

Use the **media-gateway ip** command to associate an IP address source to use for RTP traffic. When configuring VoIP, RTP traffic needs an IP address to be associated with it. However, some interfaces allow "dynamic" configuration of IP addresses, and thus, this value could change periodically. Use the **no** form of these commands to disable these functions.Variations of this command include:

**media-gateway ip loopback** *<ip address>*
**media-gateway ip primary**
**media-gateway ip secondary** *<ip address>*

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies a valid IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| **loopback** | Use an IP address statically defined to a loopback interface. Helpful when using a single IP address across multiple WAN interfaces for RTP traffic. |
| **primary** | Use the IP address that is configured as primary on this interface. Applies to static, DHCP, or negotiated addresses. |
| **secondary** *<ip-address>* | Use the statically defined secondary IP address of this interface to be used for RTP traffic. |

## Default Values

By default, **media-gateway ip** is disabled.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example configures the unit to use the primary IP address for RTP traffic:

(config)**#interface vlan 1**
(config-interface-vlan 1)**#media-gateway ip primary**

# qos-policy out *<name>*

Use the **qos-policy out** command to apply a previously-configured QoS map to outgoing packets on an interface. Use the **no** form of this command to remove the map from the interface.

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies the name of a previously-created QoS map (refer to *qos map <name> <number>* on page 643 for more information). |

## Default Values

No default value is necessary for this command.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Functional Notes

When a QoS policy is applied to an interface, it may be disabled if the interface bandwidth is not adequate to support the requested bandwidth on the map set. Once the bandwidth problem is resolved, the map will work again. The bandwidth will be rechecked on any of the following changes:

1. A priority or class-based entry is added to, deleted from, or changed in a QoS map set.
2. The interface bandwidth is changed by the **bandwidth** command on the interface.
3. A QoS policy is applied to an interface.
4. A cross-connect is created that includes an interface with a QoS policy.
5. The interface queuing method is changed to fair-queue to use weighted fair queuing.
6. The interface operational status changes.
7. The interface bandwidth changes for other reasons (e.g., when ADSL finishes training).

In order to prevent the map from being disabled in cases of temporary inadequate bandwidth (e.g., a single link goes down in a dual T1 multilink configuration where the map requests more than one T1's worth of bandwidth), the QoS map uses the maximum theoretical bandwidth on an interface, not the actual bandwidth at that time. This actually helps QoS keep higher priority class-based traffic working better than best-effort traffic when the bandwidth drops.

## Usage Examples

The following example applies the QoS map **VOICEMAP** to the VLAN interface:

(config)**#interface vlan 1**
(config-interface-vlan 1)**#qos-policy out VOICEMAP**

# snmp trap

Use the **snmp trap** command to enable all supported Simple Network Management Protocol (SNMP) traps on the interface. Use the **no** form of this command to disable this trap.

## Syntax Description

No subcommands.

## Default Values

By default, all interfaces (except virtual Frame Relay interfaces and sub-interfaces) have SNMP traps enabled.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 3.1 | Command was extended to the SHDSL interface. |
| Release 5.1 | Command was expanded to include Ethernet sub-interfaces and Gigabit Ethernet interfaces. |
| Release 6.1 | Command was expanded to include VLAN interfaces. |

## Usage Examples

The following example enables SNMP capability on the VLAN interface:

(config)**#interface vlan 1**
(config-interface-vlan 1)**#snmp trap**

# snmp trap link-status

Use the **snmp trap link-status** command to control the Simple Network Management Protocol (SNMP) variable ifLinkUpDownTrapEnable (RFC 2863) to enable (or disable) the interface to send SNMP traps when there is an interface status change. Use the **no** form of this command to disable this trap.

## Syntax Description

No subcommands.

## Default Values

By default, the ifLinkUpDownTrapEnable OID is set to enabled for all interfaces except virtual Frame Relay interfaces.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |
| Release 3.1 | Command was extended to the SHDSL interface. |
| Release 5.1 | Command was expanded to include Gigabit Ethernet, port channel, VLAN, E1, and G.703 interfaces. |
| Release 6.1 | Command was expanded to include VLAN interfaces. |

## Functional Notes

The **snmp trap link-status** command is used to control the RFC 2863 ifLinkUpDownTrapEnable OID (OID number 1.3.6.1.2.1.31.1.1.1.14.0).

## Usage Examples

The following example disables the link-status trap on the VLAN interface:

(config)**#interface vlan 1**
(config-interface-vlan 1)**#no snmp trap link-status**

# traffic-shape rate <*value*>

Use the **traffic-shape rate** command to specify and enforce an output bandwidth for the VLAN interface. Use the **no** form of this command to disable this feature. Variations of this command include:

**traffic-shape rate** <*value*>
**traffic-shape rate** <*value*> <*burst*>

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the rate (in bits per second) at which the interface should be shaped. |
| <*burst*> | Optional. Specifies the allowed burst in bytes. By default, the burst is specified as the rate divided by 5 and represents the number of bytes that would flow within 200 ms. |

## Default Values

By default, **traffic-shape rate** is disabled.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Functional Notes

Traffic shaping can be used to limit the VLAN interface to a particular rate or to specify use of QoS.

## Usage Examples

The following example sets the outbound rate of **vlan 1** to 128 kbps and applies a QoS policy that all RTP traffic is given priority over all other traffic:

(config)**#qos map voip 1**
(config-qos-map)**#match ip rtp 10000 10500 all**
(config-qos-map)**#priority unlimited**
(config-qos-map)**#interface vlan 1**
(config-interface-vlan 1)**#traffic-shape rate 128000**
(config-interface-vlan 1)**#qos-policy out voip**

# CA PROFILE CONFIGURATION COMMAND SET

To activate the Certificate Authority (CA) Profile Configuration mode, enter the **crypto ca profile** command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#crypto ca profile MyProfile
Configuring New CA Profile MyProfile
(ca-profile)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:

*cross-connect on page 29*
*do on page 33*
*end on page 34*
*exit on page 35*

All other commands for this command set are described in this section in alphabetical order.



# crl optional

Use the **crl optional** command to make CRL verification optional.

## Syntax Description

No subcommands.

## Default Values

By default, CRL optional is enabled.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

If enabled, the AOS is able to accept certificates even if no CRL is loaded into the configuration. Currently, this is the only mode supported by the AOS for CRL negotiations.

## Usage Examples

The following example sets CRL verification as optional:

(ca-profile)**#crl optional**

# email address *<address>*

Use the **email address** command to specify that an email address should be included in the certificate request. Use the **no** form of this command to remove an email address.

## Syntax Description

| | |
|---|---|
| *<address>* | Specifies the complete email address to use when sending certificate requests. This field allows up to 51 characters. |

## Default Values

No defaults necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

Configuring this setting simplifies the **crypto ca enroll** dialog, allowing you to enter the email address only once rather than every time you go through the enrollment process. Refer to *crypto ca enroll <name>* on page 463.

## Usage Examples

The following example specifies **joesmith@company.com** as the email address to be sent in certificate requests:

(ca-profile)**#email address joesmith@company.com**

# enrollment retry

Use the **enrollment retry** command to determine how the AOS handles certificate requests. Use the **no** form of this command to return to the default settings. Variations of this command include:

**enrollment retry count** <*number*>
**enrollment retry period** <*value*>

## Syntax Description

| | |
|---|---|
| **count** <*number*> | Specifies the number of times the AOS re-sends a certificate request when it does not receive a response from the previous request. Range is 1 to 100. |
| **period** <*value*> | Specifies the time period between certificate request retries. The default is 1 minute between retries. Range is 1 to 60 minutes. |

## Default Values

By default, period is set to 5 minutes, and count is set to 12 retries.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following example configures the AOS to send certificate requests every 2 minutes, stopping after 50 retries (if no response is received):

(ca-profile)**#enrollment retry count 50**
(ca-profile)**#enrollment retry period 2**

# enrollment terminal

Use the **enrollment terminal** command to specify manual (i.e., cut-and-paste) certificate enrollment. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, this command is enabled.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

This mode is overridden if the **enrollment url** command specifies the CA to which automatic certificate requests are to be sent via simple certificate exchange protocol (SCEP). Issuing an **enrollment terminal** command after using the **enrollment url** command deletes the URL and forces the unit to use manual enrollment. Refer to *enrollment url <url>* on page 1591 for more information.

## Usage Examples

The following example configures the AOS to accept manual certificate enrollment input:

(ca-profile)**#enrollment terminal**

# enrollment url <*url*>

Use the **enrollment url** command to specify the URL of the CA to which the AOS should send certificate requests. Use the **no** form of this command to remove a URL.

## Syntax Description

| | |
|---|---|
| <*url*> | Specifies the certificate authority's URL (for example, **http://10.10.10.1:400/abcdefg/pkiclient.exe**). |

## Default Values

No defaults necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

When entering the URL **http://** is required, followed by the IP address or DNS name of the CA. If the port number is something other than 80, include it after the IP address or DNS name separated with a colon (**:**).

The CA may have other necessary information to include in the CGI path before ending with the actual CGI program. An example template to follow is **http://hostname:port/path/to/program.exe**.

Use the default program **pkiclient.exe** without specifying it, end the URL with a slash (**/**). Otherwise, you must enter the program name to use. For example, **http://10.10.10.1:400/abcdefg/** will assume **pkiclient.exe** as the program (but not including the terminating slash is a configuration error).

Specifying this command will override the **enrollment terminal** setting as described previously (refer to *enrollment terminal* on page 1590).

## Usage Examples

The following example specifies **http://CAserver/certsrv/mscep/mscep.dll** as the URL to which the AOS will send certificate requests:

(ca-profile)**#enrollment url http://CAserver/certsrv/mscep/mscep.dll**

# fqdn *<name>*

Use the **fqdn** command to specify a fully-qualified domain name (FQDN) to be included in the certificate requests. Use the **no** form of this command to remove an FQDN.

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies the FQDN (e.g., company.com) to be included in requests. |

## Default Values

No defaults necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

Configuring this setting simplifies the **crypto ca enroll** dialog, allowing you to enter the FQDN only once rather than every time you go through the enrollment process. Refer to *crypto ca enroll <name>* on page 463.

## Usage Examples

The following example specifies **company.com** as the FQDN to be sent in certificate requests:

(ca-profile)**#fqdn company.com**

# ip-address *<ip address>*

Use the **ip-address** command to specify an IP address to be included in the certificate requests. Use the **no** form of this command to remove a defined IP address.

## Syntax Description

| | |
|---|---|
| *<address>* | Specifies a valid IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

## Default Values

No defaults necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

Configuring this setting simplifies the **crypto ca enroll** dialog, allowing you to enter the IP address only once rather than every time you go through the enrollment process. Refer to *crypto ca enroll <name>* on page 463.

## Usage Examples

The following example specifies **66.203.52.193** as the IP address to be sent in certificate requests:

(ca-profile)**#ip-address 66.203.52.193**

# password *<password>*

Use the **password** command to specify the challenge password for simple certificate exchange protocol (SCEP). Use the **no** form of this command to allow CA requests to be sent automatically (using SCEP) without requiring a password.

## Syntax Description

| | |
|---|---|
| *<password>* | Specifies the SCEP password (up to 80 characters). |

## Default Values

By default, no password is required.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

There are two places for configuring a SCEP password:

- At the **(ca-profile)#** prompt.
- If it is not configured at the **(ca-profile)#** prompt, you are prompted to enter one when going through the certificate enrollment process.

The password is sent to the CA from which you are requesting a certificate. The CA may then ask for the password later before a certificate can be revoked. Refer to *crypto ca enroll <name>* on page 463.

## Usage Examples

The following example sets the SCEP challenge password to **adtran**:

(ca-profile)**#password adtran**

# serial-number

Use the **serial-number** command to specify that a serial number will be included in the certificate request. Use the **no** form of this command to prevent a serial number from being included in the certificate request.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

By default, this command is set to **no serial-number**, which means that the serial number is not included in the certificate requests.

## Usage Examples

The following example configures AOS to include a serial number in the certificate request:

(ca-profile)**#serial-number**

# subject-name *<name>*

Use the **subject-name** command to specify the subject name used in the certificate request. Use the **no** form of this command to remove a configured subject name.

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies a subject name string using up to 256 characters entered in X.500 LDAP format. |

## Default Values

By default, there is no subject name configured.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

Configuring this setting simplifies the **crypto ca enroll** dialog, allowing you to enter the subject name only once rather than every time you go through the enrollment process. Refer to *crypto ca enroll <name>* on page 463.

## Usage Examples

The following example assigns a subject name of **Adtran-cert** to certificate requests:

(ca-profile)**#subject-name Adtran-cert**

# CERTIFICATE CONFIGURATION COMMAND SET

To activate the Certificate Configuration mode, enter the **crypto ca certificate chain** command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#crypto ca certificate chain MyProfile
(config-cert-chain)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:

*cross-connect on page 29*
*do on page 33*
*end on page 34*
*exit on page 35*

All other commands for this command set are described in this section in alphabetical order.

*certificate <serial number> on page 1598*
*certificate ca <serial number> on page 1599*
*crl on page 1600*

# **certificate** *<serial number>*

Use the **certificate** command to restore a certificate. Use the **no** form of this command to remove a specific certificate from the certificate chain.

## Syntax Description

| | |
|---|---|
| *<serial number>* | Specifies the certificate's serial number (up to 51 characters). This value can be found for existing certificates by using the **show run** command. |

## Default Values

No defaults necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

The user typically does not enter this command. It is primarily used to restore certificates from the startup configuration when the product is powered up.

## Usage Examples

The following example removes the certificate with the serial number **73f0bfe5ed8391a54d1214390a36cee7**:

(config)**#crypto ca certificate chain MyProfile**
(config-cert-chain)**#no certificate 73f0bfe5ed8391a54d1214390a36cee7**

# certificate ca *<serial number>*

Use the **certificate ca** command to restore a CA certificate. Use the **no** form of this command to remove a specific certificate from the certificate chain for a CA.

## Syntax Description

| | |
|---|---|
| *<serial number>* | Specifies the certificate's serial number (up to 51 characters). This value can be found for existing certificates by using the **show run** command. |

## Default Values

No defaults necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

The user typically does not enter this command. It is primarily used to restore certificates from the startup configuration when the product is powered up.

## Usage Examples

The following example removes the CA certificate with the serial number **0712**:

(config)**#crypto ca certificate chain MyProfile**
(config-cert-chain)**#no certificate ca 0712**

# crl

Use the **crl** command to restore a CRL. Use the **no** form of this command to remove the CRL for the specific CA.

## Syntax Description

No subcommands.

## Default Values

No defaults necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

The user typically does not enter this command. It is primarily used to restore CRLs from the startup configuration when the product is powered up.

## Usage Examples

The following example removes the CRL for the current CA:

(config)**#crypto ca certificate chain MyProfile**
(config-cert-chain)**#no crl**

# CRYPTO MAP IKE COMMAND SET

To activate the Crypto Map IKE mode, enter a valid version of the **crypto map ipsec-ike** command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#crypto map Map-Name 10 ipsec-ike
(config-crypto-map)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:

*cross-connect* *on page 29*
*description <text>* *on page 32*
*do* *on page 33*
*end* *on page 34*
*exit* *on page 35*

All other commands for this command set are described in this section in alphabetical order.

*antireplay* *on page 1602*
*commit-bit* *on page 1603*
*ike-policy <number>* *on page 1605*
*match address <name>* *on page 1606*
*set peer <ip address>* *on page 1608*
*set pfs* *on page 1609*
*set security-association lifetime* *on page 1610*
*set transform-set* *on page 1611*


*For VPN configuration example scripts, refer to the technical support note **Configuring VPN** located on the ADTRAN OS System Documentation CD provided with your unit.*

# antireplay

Use the **antireplay** command to enable anti-replay sequence number checking for all security associations created on this crypto map. Use the **no** form of this command to disable. Variations of this command include:

**antireplay**
**antireplay** <*value*>

## Syntax Description

| | |
|---|---|
| <*value*> | Optional. Specifies the anti-replay window size in bytes. Select from 64,128, 256, 512, or 1024 bytes. |

## Default Values

By default, the window size is set to 64 bytes.

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |

## Usage Examples

The following example enables anti-replay sequence checking on crypto map **VPN 100**:

(config)**#crypto map VPN 100 ipsec-ike**
(config-crypto-map)**#antireplay**

# commit-bit

Use the **commit-bit** command to set the commit-bit in the Internet Security Association and Key Management Protocol (ISAKMP) header when sending the second message of quick mode on an IPSec tunnel negotiation. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, the commit-bit will be used.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Functional Notes

As an extra security measure, the commit-bit can be set by the responder of a quick mode negotiation to force the initiator to wait for the fourth message of quick mode before bringing up its IPSec security associations (SA's). By default, this feature is enabled on all AOS products with VPN capabilities. Some vendors, however, may have incorrect implementations of the commit-bit that do not interoperate well with AOS products. In that case, the commit-bit should be disabled on all crypto maps that have a peer which does not support the commit-bit.

## Usage Example

The following example disables the use of commit-bit:

```
(config-crypto-map)#no commit-bit
```

The following example displays a configuration with the commit-bit disabled:

```
ip crypto
!
crypto ike policy 100
  initiate main
  respond main
  local-id address 10.10.10.1
  peer 192.168.1.1
  attribute 2
   encryption aes-256-cbc
   authentication pre-share
   lifetime 3600
!
```

```
crypto ike remote-id address 10.10.10.1 preshared-key adtran ike-policy 100 crypto map VPN 10
     no-mode-config no-xauth
!
crypto ipsec transform-set esp-aes-256-cbc-esp-sha-hmac esp-aes-256-cbc esp-sha-hmac
     mode tunnel
!
crypto map VPN 10 ipsec-ike
  description VPN to Main Site
  match address VPN-10-vpn-selectors
  set peer 192.168.1.1
  set transform-set esp-aes-256-cbc-esp-sha-hmac
  set security-association lifetime seconds 3600
  no commit-bit
  ike-policy 100
```

# ike-policy *<number>*

Use the **ike-policy** command to ensure that only a specified IKE policy is used to establish the IPSec tunnel. This prevents any mobile VPN policies from using IPSec policies that are configured for static VPN peer policies. Use the **no** form of this command to remove a configured policy.

## Syntax Description

| | |
|---|---|
| *<number>* | Specifies the policy number of the policy to assign to this crypto map. |

## Default Values

No defaults necessary for this command.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Usage Examples

The following example shows a typical crypto map configuration:

(config)**#crypto ike policy 100**
(config)**#crypto map VPN 10 ipsec-ike**
(config-crypto-map)**#description "Remote Office"**
(config-crypto-map)**#match address VPN-10-vpn-selectors**
(config-crypto-map)**#set peer 10.22.17.13**
(config-crypto-map)**#set transform-set esp-3des-esp-md5-hmac**
(config-crypto-map)**#ike-policy 100**

# match address <*name*>

Use the **match address** command to assign an IP access control list to a crypto map definition. The access control list designates the IP packets to be encrypted by this crypto map. Use the **no** form of this command to delete an IP access control list. Refer to *ip access-list extended <name>* on page 492 for more information on creating access control lists.

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies the name of the access control list you wish to assign to this crypto map. |

## Default Values

By default, no IP access control lists are defined.

## Command History

| | |
|---|---|
| Release 4.1 | Command was introduced. |

## Functional Notes

Crypto map entries do not directly contain the selectors used to determine which data to secure. Instead, the crypto map entry refers to an access control list. An access control list (ACL) is assigned to the crypto map using the **match address** command. If no ACL is configured for a crypto map, then the entry is incomplete and will have no effect on the system.

The entries of the ACL used in a crypto map should be created with respect to traffic sent by the ADTRAN product. The source information must be the local ADTRAN product and the destination must be the peer.

Only extended access control lists can be used in crypto maps.

## Usage Examples

The following example shows setting up an ACL (called **NewList**) and then assigning the new list to a crypto map (called **NewMap**):

(config)**#ip access-list extended NewList**

Configuring New Extended ACL “NewList”
(config-ext-nacl)**#exit**
(config)**#crypto map NewMap 10 ipsec-ike**
(config-crypto-map)**#match address NewList**

## Technology Review

A crypto map entry is a single policy that describes how certain traffic is to be secured. There are two types of crypto map entries: **ipsec-manual** and **ipsec-ike**. Each entry is given an index, which is used to sort the ordered list.

When a nonsecured packet arrives on an interface, the crypto map set associated with that interface is processed in order. If a crypto map entry matches the nonsecured traffic, the traffic is discarded.

When a packet is to be transmitted on an interface, the crypto map set associated with that interface is processed in order. The first crypto map entry that matches the packet will be used to secure the packet. If a suitable SA exists, that is used for transmission. Otherwise, IKE is used to establish an SA with the peer. If no SA exists, and the crypto map entry is “respond only,” the packet is discarded.

When a secured packet arrives on an interface, its SPI is used to look up an SA. If an SA does not exist, or if the packet fails any of the security checks (bad authentication, traffic does not match SA selectors, etc.), it is discarded. If all checks pass, the packet is forwarded normally.

# set peer *<ip address>*

Use the **set peer** command to set the IP address of the peer device. This must be set for multiple remote peers. Use the **no** form of this command to remove a peer device.

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies the IP address of the peer device. If this is not configured, it implies responder only to any peer. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

## Default Values

There are no default settings for this command.

## Command History

| | |
|---|---|
| Release 4.1 | Command was introduced. |

## Functional Notes

If no peer IP addresses are configured, the entry will only be used to respond to IPSec requests; it cannot initiate the requests (since it doesn't know which IP address to send the packet to). If a single peer IP address is configured, the crypto map entry can be used to both initiate and respond to SAs.

The peer IP address is the public IP address of the device which will terminate the IPSec tunnel. If the peer IP address is not static, the ADTRAN product cannot initiate the VPN tunnel. By setting no peer IP address, the ADTRAN product can respond to an IPSec tunnel request.

## Usage Examples

The following example sets the peer IP address of 10.100.23.64:

(config-crypto-map)**#set peer 10.100.23.64**

# set pfs

Use the **set pfs** command to choose the type of perfect forward secrecy (if any) that will be required during IPSec negotiation of security associations for this crypto map. Use the **no** form of this command to require no PFS. Variations of this command include:

**set pfs group1**
**set pfs group2**

## Syntax Description

| | |
|---|---|
| **group1** | Requires IPSec to use Diffie-Hellman Group 1 (768-bit modulus) exchange during IPSec SA key generation. |
| **group2** | Requires IPSec to use Diffie-Hellman Group 2 (1024-bit modulus) exchange during IPSec SA key generation. |

## Default Values

By default, no PFS will be used during IPSec SA key generation.

## Command History

| | |
|---|---|
| Release 4.1 | Command was introduced. |

## Functional Notes

If left at the default setting, no perfect forward secrecy (PFS) will be used during IPSec SA key generation. If PFS is specified, then the specified Diffie-Hellman Group exchange will be used for the initial and all subsequent key generation, thus providing no data linkage between prior keys and future keys.

## Usage Examples

The following example specifies use of the Diffie-Hellman Group 1 exchange during IPSec SA key generation:

(config-crypto-map)**#set pfs group 1**

# set security-association lifetime

Use the **set security-association lifetime** command to define the lifetime (in kilobytes and/or seconds) of the IPSec SAs created by this crypto map. Use the **no** form of this command to return to the default settings. Variations of this command include:

**set security-association lifetime kilobytes** <*value*>
**set security-association lifetime seconds** <*value*>

## Syntax Description

| | |
|---|---|
| **kilobytes** <*value*> | Specifies the SA lifetime limit in kilobytes. |
| **seconds** <*value*> | Specifies the SA lifetime limit in seconds. |

## Default Values

By default, the **security-association lifetime** is set to 28,800 seconds and there is no default for the kilobytes lifetime.

## Command History

| | |
|---|---|
| Release 4.1 | Command was introduced. |

## Functional Notes

Values can be entered for this command in both kilobytes and seconds. Whichever limit is reached first will end the security association.

## Usage Examples

The following example sets the SA lifetime to **300** kilobytes and 2 hours (**7200** seconds):

(config-crypto-map)**#set security-association lifetime kilobytes 300**
(config-crypto-map)**#set security-association lifetime seconds 7200**

# set transform-set

Use the **set transform-set** command to assign up to six transform sets to a crypto map. Use the **no** form of this command to return to the default setting. Refer to *crypto ipsec transform-set <name> <parameters>* on page 474 for information on defining transform sets. Variations of this command include:

**set transform-set** *<name>*
**set transform-set** *<name> <name>*
**set transform-set** *<name> <name> <name>*
**set transform-set** *<name> <name> <name> <name>*
**set transform-set** *<name> <name> <name> <name> <name>*
**set transform-set** *<name> <name> <name> <name> <name> <name>*

## Syntax Description

| | |
|---|---|
| *<name>* | Assigns up to six transform sets to this crypto map by listing the set names, separated by a space. |

## Default Values

By default, there is no transform set assigned to the crypto map.

## Command History

| | |
|---|---|
| Release 4.1 | Command was introduced. |

## Functional Notes

Crypto map entries do not directly contain the transform configuration for securing data. Instead, the crypto map is associated with transform sets which contain specific security algorithms (refer to *crypto ipsec transform-set <name> <parameters>* on page 474).

If no transform set is configured for a crypto map, then the entry is incomplete and will have no effect on the system.

## Usage Examples

The following example first creates a transform set (**Set1**) consisting of two security algorithms (up to three may be defined), and then assigns the transform set to a crypto map (**Map1**):

(config)**#crypto ipsec transform-set Set1 esp-3des esp-sha-hmac**
(cfg-crypto-trans)**#exit**
(config)**#crypto map Map1 1 ipsec-ike**
(config-crypto-map)**#set transform-set Set1**

# CRYPTO MAP MANUAL COMMAND SET

To activate the Crypto Map Manual mode, enter a valid version of the **crypto map ipsec-manual** command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#crypto map Map-Name 10 ipsec-manual
(config-crypto-map)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:

*description <text>* *on page 32*
*cross-connect* *on page 29*
*do* *on page 33*
*end* *on page 34*
*exit* *on page 35*

All other commands for this command set are described in this section in alphabetical order.

*antireplay* *on page 1613*
*ike-policy <number>* *on page 1614*
*match address <name>* *on page 1615*
*set peer <ip address>* *on page 1617*
*set session-key* *on page 1618*
*set transform-set <name>* *on page 1622*


*For VPN configuration example scripts, refer to the technical support note **Configuring VPN** located on the ADTRAN OS System Documentation CD provided with your unit.*

# antireplay

Use the **antireplay** command to enable anti-replay sequence number checking for all security associations created on this crypto map. Use the **no** form of this command to disable. Variations of this command include:

**antireplay**
**antireplay** <*value*>

## Syntax Description

| | |
|---|---|
| <*value*> | Optional. Specifies the anti-replay window size in bytes. Select from 64,128, 256, 512, or 1024 bytes. |

## Default Values

By default, the window size is set to 64 bytes

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |

## Usage Examples

The following example enables anti-replay sequence checking on crypto map **VPN 100**:

(config)**#crypto map VPN 100 ipsec-manual**
(config-crypto-map)**#antireplay**

# ike-policy *<number>*

Use the **ike-policy** command to ensure that only a specified IKE policy is used to establish the IPSec tunnel. This prevents any mobile VPN policies from using IPSec policies that are configured for static VPN peer policies. Use the **no** form of this command to remove an configured IKE policy.

## Syntax Description

| | |
|---|---|
| *<number>* | Specifies the policy number of the policy to assign to this crypto map. |

## Default Values

No defaults necessary for this command.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Usage Examples

The following example shows a typical crypto map configuration:

(config)**#crypto ike policy 100**
(config)**#crypto map VPN 10 ipsec-manual**
(config-crypto-map)**#description "Remote Office"**
(config-crypto-map)**#match address VPN-10-vpn-selectors**
(config-crypto-map)**#set peer 10.22.17.13**
(config-crypto-map)**#set transform-set esp-3des-esp-md5-hmac**
(config-crypto-map)**#ike-policy 100**

# match address *<name>*

Use the **match address** command to assign an IP access control list to a crypto map definition. The access control list designates the IP packets to be encrypted by this crypto map. Use the **no** form of this command to remove a defined IP access control list. Refer to *ip access-list extended <name>* on page 492 for more information on creating access control lists.

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies the name of the access control list you wish to assign to this crypto map. |

## Default Values

By default, no IP access control lists are defined.

## Command History

| | |
|---|---|
| Release 4.1 | Command was introduced. |

## Functional Notes

Crypto map entries do not directly contain the selectors used to determine which data to secure. Instead, the crypto map entry refers to an access control list. An access control list (ACL) is assigned to the crypto map using the **match address** command (refer to *crypto map* on page 475) with the NetVanta 2000 and 3000 Series units. If no ACL is configured for a crypto map, then the entry is incomplete and will have no effect on the system.

The entries of the ACL used in a crypto map should be created with respect to traffic sent by the ADTRAN product. The source information must be the local ADTRAN product, and the destination must be the peer.

Only extended access control lists can be used in crypto maps.

## Usage Examples

The following example shows setting up an access control list (called **NewList**) and then assigning the new list to a crypto map (called **NewMap**):

(config)**#ip access-list extended NewList**

Configuring New Extended ACL "NewList"

(config-ext-nacl)**#exit**
(config)**#crypto map NewMap 10 ipsec-manual**
(config-crypto-map)**#match address NewList**

## Technology Review

A crypto map entry is a single policy that describes how certain traffic is to be secured. There are two types of crypto map entries: **ipsec-manual** and **ipsec-ike**. Each entry is given an index, which is used to sort the ordered list.

When a nonsecured packet arrives on an interface, the crypto map set associated with that interface is processed in order. If a crypto map entry matches the nonsecured traffic, the traffic is discarded.

When a packet is to be transmitted on an interface, the crypto map set associated with that interface is processed in order. The first crypto map entry that matches the packet will be used to secure the packet. If a suitable SA exists, that is used for transmission. Otherwise, IKE is used to establish an SA with the peer. If no SA exists, and the crypto map entry is “respond only,” the packet is discarded.

When a secured packet arrives on an interface, its SPI is used to look up an SA. If an SA does not exist, or if the packet fails any of the security checks (bad authentication, traffic does not match SA selectors, etc.), it is discarded. If all checks pass, the packet is forwarded normally.

# set peer *<ip address>*

Use the **set peer** command to set the IP address of the peer device. Use the **no** form of this command to remove a peer device.

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies the IP address of the peer device. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

## Default Values

There are no default settings for this command.

## Command History

| | |
|---|---|
| Release 4.1 | Command was introduced. |

## Functional Notes

If no peer IP address is configured, the manual crypto map is not valid and not complete. A peer IP address is required for manual crypto maps. To change the peer IP address, the **no set peer** command must be issued first; then the new peer IP address can be configured.

## Usage Examples

The following example sets the peer IP address of **10.100.23.64**:

(config-crypto-map)**#set peer 10.100.23.64**

# set session-key

Use the **set session-key** command to define the encryption and authentication keys for this crypto map. Use the **no** form of this command to remove defined encryption and authentications keys. Variations of this command include the following:

**set session-key inbound ah** <*SPI*> <*key*>
**set session-key inbound esp** <*SPI*> **authenticator** <*key*>
**set session-key inbound esp** <*SPI*> **cipher** <*key*>
**set session-key inbound esp** <*SPI*> **cipher** <*key*> **authenticator** <*key*>
**set session-key outbound ah** <*SPI*> <*key*>
**set session-key outbound esp** <*SPI*> **authenticator** <*key*>
**set session-key outbound esp** <*SPI*> **cipher** <*key*>
**set session-key outbound esp** <*SPI*> **cipher** <*key*> **authenticator** <*key*>

## Syntax Description

| | |
|---|---|
| **inbound** | Defines encryption keys for inbound traffic. |
| **outbound** | Defines encryption keys for outbound traffic. |
| **ah** <*SPI*> | Specifies authentication header protocol and security parameter index (SPI). |
| **esp** <*SPI*> | Specifies encapsulating security payload protocol and SPI. |
| **cipher** <*key*> | Specifies encryption/decryption key. |
| **authenticator** <*key*> | Specifies authentication key. |

## Default Values

There are no default settings for this command.

## Command History

| | |
|---|---|
| Release 4.1 | Command was introduced. |

## Functional Notes

The inbound local security parameter index (SPI) must equal the outbound remote SPI. The outbound local SPI must equal the inbound remote SPI. The key values are the hexadecimal representations of the keys. They are not true ASCII strings. Therefore, a key of 3031323334353637 represents 01234567.

Refer to the following list for key length requirements.

| Algorithm: | Minimum key length required: |
|---|---|
| DES | 64-bits in length; 8 hexadecimal bytes |
| 3DES | 192-bits in length; 24 hexadecimal bytes |
| AES-128-CBC | 128-bits in length; 16 hexadecimal bytes |
| AES-192-CBC | 192-bits in length; 24 hexadecimal bytes |
| AES-256-CBC | 256-bits in length; 32 hexadecimal bytes |
| MD5 | 128-bits in length; 16 hexadecimal bytes |
| SHA1 | 160-bits in length; 20 hexadecimal bytes |

## Technology Review

The following example configures an AOS product for VPN using IPSec manual keys. This example assumes that the AOS product has been configured with a WAN IP Address of 63.97.45.57 on interface **ppp 1** and a LAN IP Address of 10.10.10.254 on interface **ethernet 0/1**. The Peer Private IP Subnet is 10.10.20.0.

For more detailed information on VPN configuration, refer to the technical support note *Configuring VPN* located on the *ADTRAN OS System Documentation* CD provided with your unit.

Step 1:
Enter the Global Configuration mode (i.e., config terminal mode).

>**enable**
**#configure terminal**

Step 2:
Enable VPN support using the **ip crypto** command. This command allows crypto maps to be applied to interfaces, and enables the IKE server to listen for IKE negotiation sessions on UDP port 500.

(config)**#ip crypto**

Step 3:
Define the transform set. A transform set defines the encryption and/or authentication algorithms to be used to secure the data transmitted over the VPN tunnel. Multiple transform sets may be defined in a system. Once a transform set is defined, many different crypto maps within the system can reference it. In this example, a transform set named **highly_secure** has been created. This transform set defines ESP with authentication implemented using 3DES encryption and SHA1 authentication.

(config)**#crypto ipsec transform-set highly_secure esp-3des esp-sha-hmac**
(cfg-crypto-trans)**#mode tunnel**

Step 4:
Define an IP access control list. An Extended Access Control List is used to specify which traffic needs to be sent securely over the VPN tunnel. The entries in the list are defined with respect to the local system. The source IP address will be the source of the traffic to be encrypted. The destination IP address will be the receiver of the data on the other side of the VPN tunnel.

(config)**#ip access-list extended corporate_traffic**
(config-ext-nacl)**#permit ip 10.10.10.0 0.0.0.255 10.10.20.0 0.0.0.255 log**
**deny ip any any**

Step 5:
Create crypto map and define manual keys. A crypto map is used to define a set of encryption schemes to be used for a given interface. A crypto map entry has a unique index within the crypto map set. The crypto map entry will specify whether IKE is used to generate encryption keys or if manually specified keys will be used. The crypto map entry will also specify who will be terminating the VPN tunnel, as well as which transform set or sets will be used to encrypt and/or authenticate the traffic on that VPN tunnel. It also specifies the lifetime of all created IPSec security associations.

The keys for the algorithms defined in the transform set associated with the crypto map will be defined by using the **set session-key** command. A separate key is needed for both inbound and outbound traffic. The key format consists of a string of hexadecimal values without the leading **0x** for each character. For example, a cipher key of **this is my cipher key** would be entered as:

**7468697320697320D7920636970686572206B6579**.

A unique Security Parameter Index (SPI) is needed for both inbound and outbound traffic. The local system's inbound SPI and keys will be the peer's outbound SPI and keys. The local system's outbound SPI and keys will be the peer's inbound SPI and keys. In this example the following keys and SPIs are used:

Inbound cipher SPI: 300 Inbound cipher key: "2te$#g89jnr(j!@4rvnfhg5e"

Outbound cipher SPI: 400 Outbound cipher key: "8564hgjelrign*&(gnb#1$d3"

Inbound authenticator key:"r5%^ughembkdhj34$x.<"

Outbound authenticator key:"io78*7gner#4(mgnsd!3"

(config)**#crypto map corporate_vpn 1 ipsec-ike**
(config-crypto-map)**#match address corporate_traffic**
(config-crypto-map)**#set peer 63.105.15.129**
(config-crypto-map)**#set transform-set highly_secure**
(config-crypto-map)**#set session-key inbound esp 300 cipher**
**32746524236738396A6E72286A21403472766E6668673565 authenticator**
**7235255E756768656D626B64686A333424782E3C**
(config-crypto-map)**#set session-key outbound esp 400 cipher**
**3835363468676A656C7269676E2A2628676E622331246433 authenticator**
**696F37382A37676E65722334286D676E73642133**

Step 6:
Configure public interface. This process includes configuring the IP address for the interface and applying the appropriate crypto map to the interface. Crypto maps are applied to the interface on which encrypted traffic will be transmitted.

(config)**#interface ppp 1**
(config-ppp 1)**#ip address 63.97.45.57 255.255.255.248**
(config-ppp 1)**#crypto map corporate_vpn**
(config-ppp 1)**#no shutdown**

Step 7:
Configure private interface to allow all traffic destined for the VPN tunnel to be routed to the appropriate gateway.

(config)**#interface ethernet 0/1**
(config-eth 0/1)**#ip address 10.10.10.254 255.255.255.0**
(config-eth 0/1)**#no shutdown**
(config-eth 0/1)**#exit**

# set transform-set *<name>*

Use the **set transform-set** command to assign a transform set to a crypto map. Use the **no** form of this command to remove assigned transform sets. Refer to *crypto ipsec transform-set <name> <parameters>* on page 474 for information on defining transform sets.

## Syntax Description

| | |
|---|---|
| *<name>* | Assigns a transform set to this crypto map by entering the set name. |

## Default Values

By default, no transform set is assigned to the crypto map.

## Command History

| | |
|---|---|
| Release 4.1 | Command was introduced. |

## Functional Notes

Crypto map entries do not directly contain the transform configuration for securing data. Instead, the crypto map is associated with transform sets which contain specific security algorithms (refer to *crypto ipsec transform-set <name> <parameters>* on page 474).

If no transform set is configured for a crypto map, then the entry is incomplete and will have no effect on the system. For manual key crypto maps, only one transform set can be specified.

## Usage Examples

The following example first creates a transform set (**Set1**) consisting of two security algorithms (up to three may be defined), and then assigns the transform set to a crypto map (**Map1**):

(config)**#crypto ipsec transform-set Set1 esp-3des esp-sha-hmac**
(cfg-crypto-trans)**#exit**

(config)**#crypto map Map1 1 ipsec-manual**
(config-crypto-map)**#set transform-set Set1**

# IKE CLIENT COMMAND SET

To activate the IKE Client mode, enter the **crypto ike client** command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#crypto ike client configuration pool ConfigPool1
(config-ike-client-pool)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:

*cross-connect* *on page 29*
*do* *on page 33*
*end* *on page 34*
*exit* *on page 35*

All other commands for this command set are described in this section in alphabetical order.

*dns-server <address1> <address2>* *on page 1624*
*ip-range <start ip address> <end ip address>* *on page 1625*
*netbios-name-server <address1> <address2>* *on page 1626*


*For VPN configuration example scripts, refer to the technical support note* ***Configuring VPN*** *located on the ADTRAN OS System Documentation CD provided with your unit.*

# dns-server <*address1*> <*address2*>

Use the **dns-server** command to specify the DNS server address(es) to assign to a client. Use the **no** form of this command to remove defined server address(es).

## Syntax Description

| | |
|---|---|
| <*address1*> | Assigns the first DNS server address. |
| <*address2*> | Optional. Assigns the second DNS server address. |

## Default Values

By default, no DNS server address is defined.

## Command History

| | |
|---|---|
| Release 4.1 | Command was introduced. |

## Usage Examples

The following example defines two DNS server addresses for this configuration pool:

(config)**#crypto ike client configuration pool ConfigPool1**
(config-ike-client-pool)**#dns-server 172.1.17.1 172.1.17.3**

# ip-range *<start ip address> <end ip address>*

Use the **ip-range** command to specify the range of addresses from which the router draws when assigning an IP address to a client. Use the **no** form of this command to remove defined IP ranges.

## Syntax Description

| | |
|---|---|
| *<start ip address>* | Specifies the first IP address in the range for this pool. |
| *<end ip address>* | Specifies the last IP address in the range for this pool.<br>IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

## Default Values

By default, no IP address range is defined.

## Command History

| | |
|---|---|
| Release 4.1 | Command was introduced. |

## Usage Examples

The following example defines an IP address range for this configuration pool:

(config)**#crypto ike client configuration pool ConfigPool1**
(config-ike-client-pool)**#ip-range 172.1.1.1 172.1.1.25**

# **netbios-name-server** *<address1> <address2>*

Use the **netbios-name-server** command to specify the NetBIOS Windows Internet Naming Service (WINS) name servers to assign to a client. Use the **no** form of this command to remove assigned name servers.

## Syntax Description

| | |
|---|---|
| *<address1>* | Specifies the first WINs server address to assign. |
| *<address2>* | Specifies the second WINs server address to assign. |

## Default Values

By default, no WINs server address is defined.

## Command History

| | |
|---|---|
| Release 4.1 | Command was introduced. |

## Usage Examples

The following example defines two WINs server addresses for this configuration pool:

(config)**#crypto ike client configuration pool ConfigPool1**
(config-ike-client-pool)**#netbios-name-server 172.1.17.1 172.1.17.25**

# IKE POLICY ATTRIBUTES COMMAND SET

To activate the IKE Policy Attributes mode, enter the **attribute** command at the IKE Policy prompt. For example:

```
>enable
#configure terminal
(config)#crypto ike policy 1
(config-ike)#attribute 10
(config-ike-attribute)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:

*cross-connect on page 29*
*do on page 33*
*end on page 34*
*exit on page 35*

All other commands for this command set are described in this section in alphabetical order.

*authentication on page 1628*
*encryption on page 1629*
*group on page 1630*
*hash on page 1631*
*lifetime <value> on page 1632*


*For VPN configuration example scripts, refer to the technical support note **Configuring VPN** located on the ADTRAN OS System Documentation CD provided with your unit.*

# authentication

Use the **authentication** command to configure this IKE policy's use of pre-shared secrets and signed certificates during IKE negotiation. Use the no form of this command to disable this feature. Variations of this command include:

**authentication dss-sig**
**authentication pre-share**
**authentication rsa-sig**

## Syntax Description

| | |
|---|---|
| **dss-sig** | Specifies to use DSS-signed certificates during IKE negotiation to validate the peer. |
| **pre-share** | Specifies the use of pre-shared secrets during IKE negotiation to validate the peer. |
| **rsa-sig** | Specifies to use RSA-signed certificates during IKE negotiation to validate the peer. |

## Default Values

By default, this command is enabled.

## Command History

| | |
|---|---|
| Release 4.1 | Command was introduced. |
| Release 5.1 | Command was expanded to include signed certificates. |

## Functional Notes

Both sides must share the same pre-shared secret in order for the negotiation to be successful.

## Usage Examples

The following example enables preshared secrets for this IKE policy:

(config-ike)**#attribute 10**
(config-ike-attribute)**#authentication pre-share**

# encryption

Use the **encryption** command to specify which encryption algorithm this IKE policy will use to transmit data over the IKE-generated SA. Use the **no** form of this command to return to the default value. Variations of this command include:

**encryption aes-128-cbc**
**encryption aes-192-cbc**
**encryption aes-256-cbc**
**encryption des**
**encryption 3des**

## Syntax Description

| | |
|---|---|
| **aes-128-cbc** | Specifies the AES-128-CBC encryption algorithm. |
| **aes-192-cbc** | Specifies the AES-192-CBC encryption algorithm. |
| **aes-256-cbc** | Specifies the AES-256-CBC encryption algorithm. |
| **des** | Specifies the DES encryption algorithm. |
| **3des** | Specifies the 3DES encryption algorithm. |

## Default Values

By default, encryption is set to DES.

## Command History

| | |
|---|---|
| Release 4.1 | Command was introduced. |

## Usage Examples

The following example selects 3DES as the encryption algorithm for this IKE policy:

(config-ike)**#attribute 10**
(config-ike-attribute)**#encryption 3des**

# group

Use the **group** command to specify the Diffie-Hellman Group (1 or 2) to be used by this IKE policy to generate the keys (which are then used to create the IPSec SA). Use the **no** form of this command to return to the default setting. Variations of this command include:

**group 1**
**group 2**

## Syntax Description

| | |
|---|---|
| **1** | Specifies 768-bit mod P. |
| **2** | Specifies1024-bit mod P. |

## Default Values

By default, group is set to 1.

## Command History

| | |
|---|---|
| Release 4.1 | Command was introduced. |

## Functional Notes

The local IKE policy and the peer IKE policy must have matching group settings in order for negotiation to be successful.

## Usage Examples

The following example sets this IKE policy to use Diffie-Hellman Group 2:

(config-ike)**#attribute 10**
(config-ike-attribute)**#group 2**

# hash

Use the **hash** command to specify the hash algorithm to be used to authenticate the data transmitted over the IKE SA. Use the **no** form of this command to return to the default setting. Variations of this command include:

**hash md5**
**hash sha**

## Syntax Description

| | |
|---|---|
| **md5** | Choose the MD5 hash algorithm. |
| **sha** | Choose the SHA hash algorithm. |

## Default Values

By default, hash is set to **sha**.

## Command History

| | |
|---|---|
| Release 4.1 | Command was introduced. |

## Usage Examples

The following example specifies **md5** as the hash algorithm:

(config-ike)**#attribute 10**
(config-ike-attribute)**#hash md5**

# lifetime <*value*>

Use the **lifetime** command to specify how long an IKE SA is valid before expiring. Use the **no** form of this command to return to the default setting.

## Syntax Description

| | |
|---|---|
| <*value*> | Specify how many seconds an IKE SA will last before expiring. |

## Default Values

By default, **lifetime** is set to 28,800 seconds.

## Command History

| | |
|---|---|
| Release 4.1 | Command was introduced. |

## Usage Examples

The following example sets a lifetime of two hours:

(config-ike)**#attribute 10**
(config-ike-attribute)**#lifetime 7200**

# IKE POLICY COMMAND SET

To activate the IKE Policy mode, enter the **crypto ike policy** command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#crypto ike policy 1
(config-ike)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:



All other commands for this command set are described in this section in alphabetical order.




*For VPN configuration example scripts, refer to the technical support note* ***Configuring VPN*** *located on the ADTRAN OS System Documentation CD provided with your unit.*

# attribute <*number*>

Use the **attribute** command to define attributes for the associated IKE policy. Multiple attributes can be created for a single IKE policy. Once you enter this command, you are in the IKE Policy Attribute mode. Refer to *IKE Policy Attributes Command Set* on page 1627 for more information. Use the **no** form of this command to remove a defined attribute.

## Syntax Description

| | |
|---|---|
| <*number*> | Assigns a number (range: 1 to 65,535) to the attribute policy. The number is the attribute's priority number and specifies the order in which the resulting VPN proposals get sent to the far end.<br>This command takes you to the (config-ike-attribute)# prompt. From here, you can configure the settings for the attribute as outlined in the section *IKE Policy Attributes Command Set* on page 1627. |

## Default Values

By default, no attribute is defined.

## Command History

| | |
|---|---|
| Release 4.1 | Command was introduced. |

## Functional Notes

Multiple attributes on an IKE policy are ordered by number (with the lowest number representing the highest priority).

## Usage Examples

The following example defines a policy attribute (**10**) and takes you into the IKE Policy Attributes:

```
(config-ike)#attribute 10
(config-ike-attribute)#
```

# client authentication host

Use the **client authentication host** command to enable the unit to act as an Xauth host when this IKE policy is negotiated with a peer. Use the **no** form of this command to disable this feature. Variations of this command include the following:

**client authentication host username** *<username>*
**client authentication host username** *<username>* **password** *<password>*
**client authentication host username** *<username>* **password** *<password>* **passphrase** *<phrase>*

## Syntax Description

| | |
|---|---|
| **password** *<password>* | Specifies the value sent via Xauth as the password. |
| **username** *<username>* | Specifies the value sent via Xauth as the username. |
| **passphrase** *<phrase>* | Optional. Specifies the value sent via Xauth as the passphrase. This is only used with authentication type OTP (one time password). |

## Default Values

By default, if this command is not present in the IKE policy the unit does not act as an Xauth host.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

The specified credentials are programmed into the unit and there is no prompt for entering values real-time. Therefore, schemes requiring real-time input or additional responses (e.g., SecureID) are not supported. The **client authentication host** command and the **client authentication server** commands are mutually exclusive. Refer to *client authentication server list <name>* on page 1637 for more information.

## Usage Examples

The following example specifies the login credentials to be sent:

(config-ike)**#client authentication host username jsmith password password1 passphrase phrase**

# client authentication host xauth-type

Use the **client authentication host xauth-type** command to allow the user to specify the Xauth authentication type if a type other than **generic** is desired. Use the **no** form of this command to return to the default setting. Variations of this command include:

**client authentication host xauth-type generic**
**client authentication host xauth-type otp**
**client authentication host xauth-type radius**

## Syntax Description

| | |
|---|---|
| **generic** | Specifies generic authentication type. |
| **otp** | Specifies OTP authentication type. |
| **radius** | Specifies RADIUS authentication type. |

## Default Values

By default, this is set to **generic**.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

This command is used along with the **client authentication host username**. Refer to *client configuration pool <name>* on page 1638 for more information. When acting as an Xauth host, this command allows the user to specify the Xauth authentication type if a type other than generic is desired.

## Usage Examples

The following example sets the Xauth type to **radius**:

(config-ike)**#client authentication host xauth-type radius**

# client authentication server list *<name>*

Use the **client authentication server list** command to enable the unit to act as an Xauth server (edge device). Use the **no** form of this command to disable this feature.

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies the named list created with the **aaa authentication login** command. |

## Default Values

By default, the router does not act as an Xauth server and extended authentication is not performed.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Functional Notes

When this IKE policy is negotiated and the peer has indicated Xauth via the IKE authentication method and/or the Xauth vendor ID, this command allows the unit to perform as an Xauth server (edge device). The specified AAA login method is used to identify the location of the user authentication database. The **client authentication host** and the **client authentication server** commands are mutually exclusive. Refer to *client configuration pool <name>* on page 1638 for more information.

## Usage Examples

The following example enables Xauth as an Xauth server and specifies which AAA method list to use in locating the user database:

(config-ike)**#client authentication server list clientusers**

# client configuration pool <*name*>

Use the **client configuration pool** command to configure the AOS to perform as mode-config server (edge device) when an IKE policy is negotiated. Use the **no** form of this command to return to the default setting. Variations of this command include the following:

**client configuration pool** <*name*>
**client configuration pool** <*name*> **initiate**
**client configuration pool** <*name*> **initiate respond**
**client configuration pool** <*name*> **respond**
**client configuration pool** <*name*> **respond initiate**

## Syntax Description

| | |
|---|---|
| <*name*> | The pool from which to obtain parameters to assign to the client. |

## Default Values

By default, if this command is not present in the IKE policy, the ADTRAN device allocates mode-config IP addresses, DNS server addresses, and NetBIOS name server addresses, and mode-config is not performed.

## Command History

| | |
|---|---|
| Release 4.1 | Command was introduced. |

## Functional Notes

This command ties an existing client configuration pool to an IKE policy.

## Usage Examples

The following example ties the **ConfigPool1** configuration pool to this IKE policy:

(config-ike)**#client configuration pool ConfigPool**

# initiate

Use the **initiate** command to allow the IKE policy to initiate negotiation (in main mode or aggressive mode) with peers. Use the **no** form of this command to allow the policy to respond only. Variations of this command include:

**initiate aggressive**
**initiate main**

## Syntax Description

| | |
|---|---|
| **aggressive** | Specifies to initiate using aggressive mode. Aggressive mode can be used when one end of the VPN tunnel has a dynamically assigned address. The side with the dynamic address must be the initiator of the traffic and tunnel. The side with the static address must be the responder. |
| **main** | Specifies to initiate using main mode. Main mode requires that each end of the VPN tunnel has a static WAN IP address. Main mode is more secure than aggressive mode because more of the main mode negotiations are encrypted. |

## Default Values

By default, the **main** initiation mode is enabled.

## Command History

| | |
|---|---|
| Release 4.1 | Command was introduced. |

## Functional Notes

By using the **initiate** and **respond** commands, you can configure the IKE policy to initiate and respond, initiate only, or respond only. It is an error if you have both **initiate** and **respond** disabled.

## Usage Examples

The following example enables the AOS device to initiate IKE negotiation in main mode:

(config-ike)**#initiate main**

# local-id

Use the **local-id** command to set the local ID for the IKE policy. This setting overrides the system local ID setting (set in the Global Configuration mode using the **crypto ike local-id address** command). Use the **no** form of this command to remove a local ID. Variations of this command include:

**local-id address** <*ip address*>
**local-id asn1-dn** <*name*>
**local-id fqdn** <*name*>
**local-id user-fqdn** <*name*>

## Syntax Description

| | |
|---|---|
| **address** <*ip address*> | Specifies a remote IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| **asn1-dn** <*name*> | Specifies an Abstract Syntax Notation Distinguished Name as the remote ID (enter this value in LDAP format). |
| **fqdn** <*name*> | Specifies a fully qualified domain name (e.g., adtran.com) as the remote ID. |
| **user-fqdn** <*name*> | Specifies a user fully qualified domain name or email address (e.g., user1@adtran.com) as the remote ID. |

## Default Values

By default, the local ID is not defined.

## Command History

| | |
|---|---|
| Release 4.1 | Command was introduced. |

## Functional Notes

The local ID for a particular IKE policy can be set in two ways. The first (default) method is done in the Global Configuration mode:

(config)**#crypto ike local-id address**

This command, which by default is executed on start-up, makes the local ID of an IKE policy equal to the IPv4 address of the interface on which an IKE negotiation is occurring. This is particularly useful for products that could have multiple public interfaces.

The second method is to use the IKE policy command:

(config-ike)**#local-id [address | fqdn | user-fqdn]** <*ip address or name*>

This policy-specific command allows you to manually set the local ID for an IKE policy on a per-policy basis. You can use both methods simultaneously in the product. Several IKE policies can be created, some of which use the default system setting of the IPv4 address of the public interface. Others can be set to override this system setting and manually configure a local ID specific to those policies. When a new IKE policy is created, they default to **no local-id**. This allows the system local ID setting to be applied to the policy.

## Usage Examples

The following example sets the local ID of this IKE policy to the IPv4 address 63.97.45.57:

(config-ike)#**local-id address 63.97.45.57**

# nat-traversal

Use the **nat-traversal** command to allow, force, or disable NAT traversal version 1 and 2 on a specific Ike policy. Use the **no** form of these commands to disable these features. Variations of this command include:

**nat-traversal v1 allow**
**nat-traversal v1 disable**
**nat-traversal v1 force**
**nat-traversal v2 allow**
**nat-traversal v2 disable**
**nat-traversal v2 force**

## Syntax Description

| | |
|---|---|
| **v1** | Specifies NAT traversal version 1. |
| **v2** | Specifies NAT traversal version 2. |
| **allow** | Sets the IKE policy to allow the specified NAT traversal version. |
| **disable** | Sets the IKE policy to disable the specified NAT traversal version. |
| **force** | Sets the IKE policy to force the specified NAT traversal version. |

## Default Values

The defaults for this command are **nat-traversal v1 allow** and **nat-traversal v2 allow**.

## Command History

| | |
|---|---|
| Release 7.1 | Command was introduced. |

## Usage Examples

The following example disables version 2 on Ike policy 1:

(config)**#crypto ike policy 1**
(config-ike)**#nat-traversal v2 disable**

# peer

Use the **peer** command to enter the IP address of the peer device. Repeat this command for multiple peers. Use the **any** keyword if you want to set up a policy that will initiate or respond to any peer. Use the **no** form of this command to remove a peer device. Variations of this command include:

**peer** <*ip address*>
**peer any**

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies a peer IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| **any** | Allows any peer to connect to this IKE policy. |

## Default Values

There are no default settings for this command.

## Command History

| | |
|---|---|
| Release 4.1 | Command was introduced. |

## Functional Notes

An IKE policy is incomplete unless one of the peer commands is specified. Only one IKE policy can be configured with **peer any**.

## Usage Examples

The following example sets multiple peers on an IKE policy for an initiate-and-respond policy using pre-shared secret, DES, MD5, and Diffie-Hellman Group 1:

(config)**#crypto ike policy 100**
(config-ike)**#peer 63.97.45.57**
(config-ike)**#peer 63.105.15.129**
(config-ike)**#peer 192.168.1.3**
(config-ike)**#respond anymode**
(config-ike)**#initiate main**

The following example sets up a policy allowing any peer to initiate using preshared secret, DES, MD5, and Diffie-Hellman Group 1.

(config)**#crypto ike policy 100**
(config-ike)**#peer any**
(config-ike)**#respond anymode**
(config-ike)**#initiate main**

## Technology Review

IKE policies must have a peer address associated with them to allow certain peers to negotiate with the ADTRAN product. This is a problem when you have “roaming” users (those who obtain their IP address using DHCP or some other dynamic means). To allow for “roaming” users, the IKE policy can be set up with **peer any** to allow any peer to negotiate with the ADTRAN product. There can only be one **peer any** policy in the running configuration.

# respond

Use the **respond** command to allow the IKE policy to respond to negotiations by a peer. Use the **no** form of this command to allow the policy to only initiate negotiations. Variations of this command include:

**respond aggressive**
**respond anymode**
**respond main**

## Syntax Description

| | |
|---|---|
| **aggressive** | Specifies to respond only to aggressive mode. |
| **anymode** | Specifies to respond to any mode. |
| **main** | Specifies to respond only to main mode. |

## Default Values

By default, respond to any mode is enabled.

## Command History

| | |
|---|---|
| Release 4.1 | Command was introduced. |

## Functional Notes

By using the **initiate** and **respond** commands, you can configure the IKE policy to initiate and respond, initiate only, or respond only. It is an error if you have both **initiate** and **respond** disabled.

## Usage Examples

The following example configures the router to initiate and respond to IKE negotiations:

(config-ike)**#respond anymode**
(config-ike)**#initiate main**

# AS PATH LIST CONFIGURATION COMMAND SET

To activate the Autonomous System (AS) Path List Configuration mode, enter the **ip as-path-list** command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#ip as-class-list listname
(config-as-path-list)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:

*cross-connect* *on page 29*
*do* *on page 33*

All other commands for this command set are described in this section in alphabetical order.

*deny <value>* *on page 1647*
*permit <value>* *on page 1648*

# deny *<value>*

Use the **deny** command to add an entry to the community list that denies BGP routes containing the specified community number in the community attribute. Use the **no** form of this command to remove the statement from the community list.

## Syntax Description

| | |
|---|---|
| *<value>* | Denies routes that contain this value in their community attribute. This is a numeric value that can be an integer from 1 to 4,294,967,295 or string in the form *aa:nn*, where *aa* is the AS number and *nn* is the community number. Multiple community number parameters can be present in the command. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example creates a community list named **MyList** to deny BGP routes that match the AS path attributes **30:22**:

(config)**#ip as-path-list MyList**
(config-comm-list)**#deny 30:22**

# **permit** <*value*>

Use the **permit** command to add an entry to the community list that allows only BGP routes containing the specified community number in the community attribute. Use the **no** form of this command to remove the statement from the community list.

## Syntax Description

| | |
|---|---|
| <*value*> | Permits routes that contain this value in their community attribute. This is a numeric value that can be an integer from 1 to 4,294,967,295 or string in the form *aa:nn*, where *aa* is the AS number and *nn* is the community number. Multiple community number parameters can be present in the command. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example creates a community list named **MyList** to permit BGP routes that match the AS path attributes **30:22**:

(config)**#ip as-path-list listname**
(config-comm-list)**#permit 30:22**

# BGP CONFIGURATION COMMAND SET

To activate the BGP Configuration mode, enter the **router bgp** command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#router bgp 1
(config-bgp)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:

*cross-connect on page 29*
*do on page 33*
*end on page 34*

All other commands for this command set are described in this section in alphabetical order.



# bgp

Use the **bgp** command to instruct AOS on how to handle Multi-Exit Discriminators (MEDs) for all routes from the same autonomous system (AS). Variations of this command include:

**bgp compare-med**
**bgp ignore-med**

## Syntax Description

| | |
|---|---|
| **compare-med** | Configures AOS to compare MEDs for all received routes. |
| **ignore-med** | Configures AOS to disregard MEDs fro all received routes. |

## Default Values

By default, AOS compares the MED attributes for routes from the same AS.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example enables this option:

(config)**#router bgp 1**
(config-bgp)**#bgp compare-med**

# bgp always-compare-med

Use the **bgp always-compare-med** command to configure AOS to always compare the Multi-Exit Discriminators (MEDs) for all paths for a route, regardless of the autonomous system (AS) through which the paths pass. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example enables this option:

(config)**#router bgp 1**
(config-bgp)**#bgp always-compare-med**

# bgp default local-preference <*value*>

Use the **bgp default local-preference** command to change the local preference for all BGP routes. The local preference is an attribute (LOCAL_PREF) that indicates a degree of preference for a route relative to other routes in the local autonomous system (AS). BGP4 neighbors can send the local preference value as an attribute of a route in an UPDATE message. Local preference only applies to routes within the local AS. Use the **no** form of this command to return to the default setting.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the local preference value. Valid range is 0 to 4,294,967,295. |

## Default Values

By default, the local preference is set to 100.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example changes the default local preference to **200**:

(config)**#router bgp 1**
(config-bgp)**#bgp local-preference 200**

# bgp deterministic-med

Use the **bgp deterministic-med** command to configure AOS to compare the Multi-Exit Discriminators (MEDs) for all BGP routes received from different neighbors within the same AS. Use the **no** form of this command to disable this option.

## Syntax Description

No subcommands.

## Default Values

By default, this option is disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example enables AOS to use the deterministic MED option:

(config)**#router bgp 1**
(config-bgp)**#bgp deterministic-med**

# bgp fast-external-failover

Use the **bgp fast-external-failover** command to enable the fast-external-failover feature. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, this command is enabled.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Functional Notes

When enabled, if the link interface over which the router is communicating with a BGP peer goes down, the BGP session with that peer is immediately cleared. When fallover is disabled and the link goes down, the session is maintained until the BGP hold timer expires.

## Usage Examples

The following example enables this option:

(config)**#router bgp 1**
(config-bgp)**#bgp fast-external-fallover**

# bgp log-neighbor-changes

Use the **bgp log-neighbor-changes** command to control the logging of neighbor state changes. Use the **no** form of this command to return to the default setting.

## Syntax Description

No subcommands.

## Default Values

By default, neighbor changes are not logged.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Functional Notes

This command controls logging of BGP neighbor state changes (up/down) and resets. This information is useful for troubleshooting and determining network stability.

## Usage Examples

The following example enables logging of BGP neighbor state changes:

(config)**#router bgp 1**
(config-bgp)**#bgp log-neighbor-changes**

# bgp router-id *<ip address>*

Use the **bgp router-id** command to specify the IP address that the router should use as its BGP router ID. Use the **no** form of this command to return to the default setting.

## Syntax Description

| | |
|---|---|
| *<ip address>* | Designates the IP address this router should use as its BGP router ID. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

## Default Values

By default, no router ID is configured. The default action is detailed in *Functional Notes*, below.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Functional Notes

This command allows an IP address to be specified for use as the BGP router ID. If no IP address is configured at BGP startup, it uses the highest IP address configured on a loopback interface. If no loopback interfaces are configured, it uses the highest IP address configured on any interface that is active. If the specified router ID is changed, existing sessions with BGP neighbors are reset.

## Usage Examples

The following example configures IP address 10.0.0.1 as the BGP router ID:

(config)**#router bgp 1**
(config-bgp)**#bgp router-id 10.0.0.1**

# distance bgp *<external> <internal> <local>*

Use the **distance bgp** command to set the administrative distance for BGP routes. Use the **no** form of this command to return to the default setting.

## Syntax Description

| | |
|---|---|
| *<external>* | Sets the administrative distance for BGP routes learned via eBGP sessions. A value of 255 means the route is not installed. Range is 1 to 254. |
| *<internal>* | Sets the administrative distance for BGP routes learned via iBGP sessions. A value of 255 means the route is not installed. Range is 1 to 254. |
| *<local>* | Sets the administrative distance for BGP routes learned via the network command and redistribution. A value of 255 means the route is not installed. Range is 1 to 254. |

## Default Values

By default external is set to 20, internal to 200, and local to 200. Normally, these default settings should not be changed.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Functional Notes

This command sets the administrative distance for BGP routes. The administrative distance is a local variable that allows a router to choose the best route when there are multiple paths to the same network. Routes with smaller administrative distances are favored.

## Usage Examples

The following example gives external BGP routes an administrative distance of **30**, internal BGP routes an administrative distance of **200**, and local routes an administrative distance of **240**:

(config)**#router bgp 1**
(config-bgp)**#distance bgp 30 200 240**

# hold-timer <*value*>

Use the **hold-timer** command to set the default hold time for all neighbors in the BGP process.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies a time interval (in seconds) within which a keepalive must be received from a peer before it is declared dead peer. Range is 0 to 65,535 seconds. |

## Default Values

By default, the hold time is 90 seconds.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Functional Notes

Using the **hold-timer** command in BGP configuration mode sets the default hold time for all neighbors in that BGP process. Using the **hold-timer** command in BGP neighbor configuration mode sets the hold time for only that neighbor. The peers will negotiate and use the lowest configured setting. The keepalive interval will be set to one third of the negotiated hold time.

## Usage Examples

The following example sets a hold time of **120** seconds for a specific neighbor, with an understood keepalive interval of 40 seconds:

(config)**#router bgp 1**
(config-bgp)**#hold-timer 120**

# maximum-paths *<value>*

Use the **maximum-paths** command to specify the number of parallel routes (shared paths) eBGP neighbors can inject into the route table. When IP load sharing is enabled, traffic is balanced to a specific destination across up to six equal paths. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| *<value>* | Specifies the number of parallel routes eBGP neighbors can inject into the route table. Valid range is 1 to 6. |

## Default Values

By default, a single path can exist in the route table.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example configures AOS to allow **4** parallel paths in the route table:

(config)**#router bgp 1**
(config-bgp)**#maximum-paths 4**

# neighbor *<ip address>*

Use the **neighbor** command to create a BGP neighbor, specify an IP address, and activate the BGP neighbor configuration commands. Refer to *BGP Neighbor Configuration Command Set* on page 1662 for more information on neighbor-specific configuration parameters. Use the **no** form of this command to remove the configured neighbor.

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies the IP address for the neighbor. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

## Default Values

By default, there are no configured BGP neighbors.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example configures a BGP neighbor with an IP address of **10.10.10.1**:

(config)**#router bgp 1**
(config-bgp)**#neighbor 10.10.10.1**
(config-bgp-neighbor)#

# **network** <*ip address*> **mask** <*subnet mask*>

Use the **network** command to add a network to the BGP network table. Use the **no** form of this command to remove the configured network.

## Syntax Description

| | |
|---|---|
| <*ip address*> | Specifies the network address for the neighbor that AOS will advertise over BGP. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| **mask** <*subnet mask*> | Specifies the subnet mask that corresponds to a range of IP addresses (network) or a specific host. Subnet masks can be expressed in dotted decimal notation (for example, **255.255.255.0**) or as a prefix length (for example, **/24**). |

## Default Values

By default, there are no configured BGP networks.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example adds the **10.10.10.1** network with a subnet mask of **255.255.255.0**:

(config)**#router bgp 1**
(config-bgp)**#network 10.10.10.1 mask 255.255.255.0**

# BGP NEIGHBOR CONFIGURATION COMMAND SET

To activate the BGP Neighbor Configuration mode, enter the **router bgp-neighbor** command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#router bgp 1
(config-bgp)#router bgp-neighbor 192.22.73.101
(config-bgp-neighbor)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:

*cross-connect on page 29*
*description <text> on page 32*
*do on page 33*
*end on page 34*
*exit on page 35*
*shutdown on page 36*

All other commands for this command set are described in this section in alphabetical order.



# advertisement-interval *<value>*

Use the **advertisement-interval** command to configure the AOS to specify how long the BGP process waits before sending updates to the neighbor. Use the **no** form of this command to return to the default setting.

## Syntax Description

| | |
|---|---|
| *<value>* | Specifies the advertisement interval in seconds. Range is 0 to 600 seconds. |

## Default Values

By default, the advertisement interval is 30 seconds for external neighbors and 5 seconds for internal neighbors.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Functional Notes

This command sets the minimum interval between sending updates to the specified neighbor.

## Usage Examples

The following example configures the BGP process to wait at least 100 seconds before sending updates to the neighbor:

(config)**#router bgp 1**
(config-bgp)**#router bgp-neighbor 192.22.73.101**
(config-bgp-neighbor)**#advertisement-interval 100**

# as-path-list <*name*>

Use the **as-path-list** command to assign a predefined autonomous system (AS) path list to a BGP neighbor. This list is then used to filter inbound and/or outbound BGP route updates. Use the **no** form of this command to discontinue use of the list. Variations of this command include:

**as-path-list** <*name*> **in**
**as-path-list** <*name*> **out**

## Syntax Description

| | |
|---|---|
| <*name*> | Assigns an AS path list to this BGP neighbor. |
| **in** | Specifies the filtering of all inbound BGP route updates. |
| **out** | Specifies the filtering of all outbound BGP route updates. |

## Default Values

By default, no AS path lists are specified for filtering.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Functional Notes

Before they can be assigned to a neighbor, AS path lists must first be defined using the **ip as-path-list** command.

## Usage Examples

The following example uses the **no15** AS path list to filter all inbound BGP route updates:

(config)**#router bgp 1**
(config-bgp)**#router bgp-neighbor 192.22.73.101**
(config-bgp-neighbor)**#as-path-list no15 in**

# distribute-list <*name*>

Use the **distribute-list** command to add route filtering functionality by assigning inbound and outbound access control lists on either a per-interface or global basis. Only one inbound/outbound pair of access control lists can be configured for a particular interface. Use the **no** form of this command to disable the filtering. Variations of this command include:

**distribute-list** <*name*> **in**
**distribute-list** <*name*> **out**

## Syntax Description

| | |
|---|---|
| <*name*> | Specifies an access control list name. This is a standard IP access control list (ACL) against which the contents of the incoming/outgoing routing updates are matched. |
| **in** | Applies route filtering to inbound data. |
| **out** | Applies route filtering to outbound data. |

## Default Values

By default, distribute-list filtering is disabled.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example will filter out all network advertisements received via Ethernet interface 0/1 with the exception of the 10.10.10.0 network:

(config)**#router bgp 1**
(config-bgp)**#router bgp-neighbor 192.22.73.101**
(config-rip)**#distribute-list list_1 in eth 0/1**
(config-rip)**#exit**
(config)**#ip access-list standard list_1**
(config-std-nacl)**#permit 10.10.10.0 0.0.0.255**

# ebgp-multihop *<value>*

Use the **ebgp-multihop** command to configure the maximum hop count of BGP messages to a neighbor. Use the **no** form of this command to return to the default setting.

## Syntax Description

| | |
|---|---|
| *<value>* | Specifies the maximum hop count of BGP messages to a neighbor. Range is 1 to 254 hops. |

## Default Values

By default, eBGP multihop is set to 1.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Functional Notes

This command allows an eBGP neighbor to be on a network that is not directly connected. Normally, eBGP peers are directly connected. In certain applications, a non-BGP device such as a firewall or router may reside between eBGP peers. In this case, the eBGP multihop command is required to allow updates to have a TTL greater than 1 and to allow received BGP updates to be added to the BGP table when the next hop address is not directly connected.

## Usage Examples

The following example allows a BGP message to travel 10 hops to a neighbor:

(config)**#router bgp 1**
(config-bgp)**#router bgp-neighbor 192.22.73.101**
(config-bgp-neighbor)**#ebgp-multihop 10**

# hold-timer <*value*>

Use the **hold-timer** command to set the default hold time for all neighbors in the BGP process. Use the **no** form of this command to return to the default setting.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies a time interval (in seconds) within which a keepalive must be received from a peer before it is declared dead peer. Range is 0 to 65,535 seconds. |

## Default Values

By default, the hold time is 90 seconds.

## Command History

| | |
|---|---|
| Release 8.1 | Command was introduced. |

## Functional Notes

Using the **hold-timer** command in BGP configuration mode sets the default hold time for all neighbors in that BGP process. Using the **hold-timer** command in BGP neighbor configuration mode sets the hold time for only that neighbor. The peers will negotiate and use the lowest configured setting. The keepalive interval will be set to one-third of the negotiated hold time.

## Usage Examples

The following example sets a hold time of **120** seconds for a specific neighbor, with an understood keepalive interval of 40 seconds:

(config)**#router bgp 1**
(config-bgp)**#router bgp-neighbor 192.22.73.101**
(config-bgp-neighbor)**#hold-timer 120**

# local-as <*value*>

Use the **local-as** command to specify an autonomous system (AS) number for the unit to use when communicating with this BGP neighbor. Use the **no** form of this command to return to default settings.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the AS number to use when communicating with this neighbor. Must be different than the AS number for this router and the peer router. Only valid for eBGP connections. Range is 0 to 65,535. |

## Default Values

By default, no local AS number is defined. The router's BGP AS number is used.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Functional Notes

This command substitutes a different AS number to be used for communicating with this BGP neighbor. (other than the one the router is actually a member of). This can be used to satisfy network designs requiring a customer to appear as one AS number when communicating with one internet service provider (ISP) and another when communicating with another ISP.

## Usage Examples

The following example configures this BGP neighbor's AS number to be **300**:

(config)**#router bgp 1**
(config-bgp)**#router bgp-neighbor 192.22.73.101**
(config-bgp-neighbor)**#local-as 300**

## Technology Review

This router appears (to the peer router) to be in the AS specified with the **local-as** command. Therefore all routes learned from the peer have this number prepended to the AS path. In network advertisements from routers using the **local-as** command, the router's true AS number (the number specified using the **router bgp as-number** command) is prepended to the AS path attribute, and the local-AS (the number specified in the **neighbor local-as** command) is prepended to the AS path attribute. This makes it appear that the path to the network is first through the local-AS, and then through the true AS. To further illustrate, consider the following example network.

In this network:

- Router A is in AS 100.
- Router B is in AS 300.
- Router A is an eBGP peer with Router B.
- Router A's connection to Router B specifies a **local-as** of 200.
- Router B is configured to connect to Router A in AS 200.

Therefore:

- To Router B, all aspects of Router A appear as AS 200.
- Networks advertised from Router A to Router B will have the AS path **200 100** prepended to the AS path attribute.
- Router A will add AS 200 to the AS path of networks learned from Router B.

# next-hop-self

Use the **next-hop-self** command to force the next hop attribute to be changed to this unit's address when advertising networks that would not have the next hop changed under normal rules. Normal next hop rules are described in the *Functional Notes* section below. Use the **no** form of this command to cause normal next hop rules to apply.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled and normal next hop rules apply.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Functional Notes

In eBGP, routes are normally advertised with a next hop set to the IP address that the receiving peer has configured in its neighbor statement for this router. In the eBGP case where the receiving router is in the same subnet as the current next hop, the current next hop is not changed.

For broadcast multiaccess networks (Ethernet), this provides more efficient routing. For non-broadcast multiaccess networks (NBMA) such as Frame Relay with a partial mesh using point-to-multipoint circuits, this rule can cause significant problems. Since the partial mesh is on the same subnet, BGP applies the rule of not changing the next hop address, rendering invalid routes in certain topologies. This is one case where this command is necessary to solve a problem.

## Usage Examples

The following example enables **next-hop-self**:

(config)**#router bgp 1**
(config-bgp)**#router bgp-neighbor 192.22.73.101**
(config-bgp-neighbor)**#next-hop-self**

# no default-originate

Use the **no default-originate** command to prevent the unit from sending the default route to a BGP neighbor. This command must be issued on a per neighbor basis.

## Syntax Description

No subcommands available for this command.

## Default Values

By default, the default route is not sent to the BGP neighbor.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example prevents the unit from sending the default route to a BGP neighbor:

(config)**#router bgp 1**
(config-bgp)**#router bgp-neighbor 192.22.73.101**
(config-bgp-neighbor)**#no default-originate**

# password *<password>*

Use the **password** command to enable MD5 password authentication on TCP. Use the **no** form of this command to disable authentication.

## Syntax Description

| | |
|---|---|
| *<password>* | Specifies the password string to be used for authentication. The password is case-sensitive and must not exceed 80 characters. |

## Default Values

By default, authentication is disabled.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Functional Notes

Authentication must be configured on both peers using the same password. Every BGP TCP segment sent is authenticated. Configuring authentication causes an existing session to be torn down and re-established using the currently specified authentication.

## Usage Examples

The following example enables authentication for this BGP neighbor and sets a password of **user1**:

(config)**#router bgp 1**
(config-bgp)**#router bgp-neighbor 192.22.73.101**
(config-bgp-neighbor)**#password user1**

# prefix-list <*name*>

Use the **prefix-list** command to assign a predefined prefix list to a BGP neighbor. The list is then used to filter BGP route updates received and/or sent from/by the specified peer. Use the **no** form of this command to discontinue use of the prefix list. Variations of this command include:

**prefix-list** <*listname*> **in**
**prefix-list** <*listname*> **out**

## Syntax Description

| | |
|---|---|
| <*name*> | Assigns the specified prefix list to this BGP neighbor. |
| **in** | Specifies that all inbound BGP route updates received from the specified peer be filtered. |
| **out** | Specifies that all outbound BGP route updates being sent to the specified peer be filtered. |

## Default Values

By default, no prefix lists are specified for filtering.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Functional Notes

Before they can be assigned to a BGP neighbor, prefix lists must first be defined using the **ip prefix-list** command. Refer to *ip prefix-list <name> description <"text">* on page 560 for more information.

## Usage Examples

The following example uses the **MyList** prefix list to filter all BGP updates received from the specified peer:

(config)**#router bgp 1**
(config-bgp)**#router bgp-neighbor 192.22.73.101**
(config-bgp-neighbor)**#prefix-list MyList in**

# remote-as <*value*>

Use the **remote-as** command to specify the BGP autonomous system (AS) to which the neighbor belongs, adding an entry to the BGP neighbor table. Use the **no** form of this command to return to default settings.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the AS number. This number must be different from the AS number of the local router (which is defined using the **router bgp** command). Range is 1 to 65,535. Refer to *router bgp* on page 651 for more information. |

## Default Values

By default, no BGP neighbors are defined.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example configures a remote AS number of **200** for this neighbor:

(config)**#router bgp 1**
(config-bgp)**#router bgp-neighbor 192.22.73.101**
(config-bgp-neighbor)**#remote-as 200**

# route-map <*name*>

Use the **route-map** command to assign a route map to this BGP neighbor. The route map is then used to filter or modify inbound and/or outbound BGP route updates. Use the **no** form of this command to return to default settings. Variations of this command include:

**route-map** <*name*> **in**
**route-map** <*name*> **out**

## Syntax Description

| | |
|---|---|
| <*name*> | Assigns the specified route map to this BGP neighbor. |
| **in** | Specifies the filtering/modification of all inbound BGP route updates. |
| **out** | Specifies the filtering/modification of all outbound BGP route updates. |

## Default Values

By default, no route map is assigned.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Functional Notes

Before a route map can be assigned to a BGP neighbor, it must first be defined using the **route-map** command. Refer to *route-map* on page 650 for more information.

## Usage Examples

The following example assigns a route map to this neighbor for outbound filtering:

(config)**#router bgp 1**
(config-bgp)**#router bgp-neighbor 192.22.73.101**
(config-bgp-neighbor)**#route-map MapName out**

# send-community standard

Use the **send-community standard** command to insert a standard BGP community attribute to all outgoing route updates for this neighbor. Use the **no** form of this command to return to default settings.

## Syntax Description

No subcommands.

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example inserts a standard BGP community attribute to all outgoing route updates for the specified neighbor:

(config)**#router bgp 1**
(config-bgp)**#router bgp-neighbor 192.22.73.101**
(config-bgp-neighbor)**#send-community standard**

# soft-reconfiguration inbound

Use the **soft-reconfiguration inbound** command to enable this unit to store BGP updates for the specified neighbor. Use the **no** form of this command to return to default settings.

## Syntax Description

No subcommands.

## Default Values

By default, this command is enabled.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Functional Notes

BGP updates are stored prior to filtering, thus allowing the **clear ip bgp soft** command to be used in the absence of route refresh (RFC2918) capability. This command affects all neighbors. Refer to *clear ip bgp* on page 56 for more information.

## Usage Examples

The following example enables the unit to store BGP updates for the specified neighbor:

(config)**#router bgp 1**
(config-bgp)**#router bgp-neighbor 192.22.73.101**
(config-bgp-neighbor)**#soft-reconfiguration inbound**

# update-source *<interface>*

Use the **update-source** command to specify which virtual interface's IP address will be used as the source IP address for the BGP TCP connection (when connecting to this peer). Use the **no** form of this command to return to default settings.

## Syntax Description

| | |
|---|---|
| *<interface>* | Specifies the virtual interface to be used as the source IP address. Specify an interface in the format *<interface type [slot/port \| slot/port.sub-interface id \| interface id \| interface id.sub-interface id]>*. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **update-source ?** for a complete list of valid interfaces. |

## Default Values

By default, the outbound interface's IP address is used for BGP updates.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Functional Notes

This is most often configured as a loopback interface that is reachable by the peer router. The peer will specify this address in its neighbor commands for this router.

## Usage Examples

The following example configures the **loopback 1** interface as the source IP:

(config)**#router bgp 1**
(config-bgp)**#router bgp-neighbor 192.22.73.101**
(config-bgp-neighbor)**#update-source loopback 1**

# COMMUNITY LIST CONFIGURATION COMMAND SET

To activate the Community List Configuration mode, enter the **ip community-list** command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#ip community-list listname
(config-comm-list)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:

*cross-connect on page 29*
*do on page 33*

All other commands for this command set are described in this section in alphabetical order.

*deny on page 1680*
*permit on page 1681*

# deny

Use the **deny** command to add an entry to the community list that denies BGP routes containing the specified community number in the community attribute. Use the **no** form of this command to remove the statement from the community list. Variations of this command include:

**deny** <*value*>
**deny internet**
**deny local-as**
**deny no-advertise**
**deny no-export**

## Syntax Description

| | |
|---|---|
| <*value*> | Denies routes that contain this value in their community attribute. This is a numeric value that can be an integer from 1 to 4,294,967,295 or string in the form *aa:nn*, where *aa* is the AS number and *nn* is the community number. Multiple community number parameters can be present in the command. |
| **internet** | Denies routes that contain the reserved community number for the INTERNET community. |
| **local-as** | Denies routes that contain the reserved community number for NO_EXPORT_SUBCONFED. Routes containing this attribute should not be advertised to external BGP peers. |
| **no-advertise** | Denies routes that contain the reserved community number for NO_ADVERTISE. Routes containing this attribute should not be advertised to any BGP peer. |
| **no-export** | Denies routes that contain the reserved community number for NO_EXPORT. Routes containing this attribute should not be advertised to BGP peers outside a confederation boundary. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example creates a community list named **MyList** to deny BGP routes that have the Internet community number in their community attribute:

(config)**#ip community-list MyList**
(config-comm-list)**#deny no-export**

# permit

Use the **permit** command to add an entry to the community list that allows only BGP routes containing the specified community number in the community attribute. Use the **no** form of this command to remove the statement from the community list. Variations of this command include:

**permit** <*value*>
**permit internet**
**permit local-as**
**permit no-advertise**
**permit no-export**

## Syntax Description

| | |
|---|---|
| <*value*> | Permits routes that contain this value in their community attribute. This is a numeric value that can be an integer from 1 to 4,294,967,295 or string in the form *aa:nn*, where *aa* is the AS number and *nn* is the community number. Multiple community number parameters can be present in the command. |
| **internet** | Permits routes that contain the reserved community number for the INTERNET community. |
| **local-as** | Permits routes that the reserved community number for NO_EXPORT_SUBCONFED. Routes containing this attribute should not be advertised to external BGP peers. |
| **no-advertise** | Permits routes that contain the reserved community number for NO_ADVERTISE. Routes containing this attribute should not be advertised to any BGP peer. |
| **no-export** | Permits routes that contain the reserved community number for NO_EXPORT. Routes containing this attribute should not be advertised to BGP peers outside a confederation boundary. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example permits BGP routes that match the AS path attributes:

(config)**#ip as-path-list listname**
(config-comm-list)**#permit 30:22**

# NETWORK MONITOR PROBE COMMAND SET

This section explains the commands available for Network Monitoring Probes. Probes are stand alone objects which help determine the status of a route based on the success or failure of probe traffic across the path. The probes can be configured to trigger at particular intervals. There are three types of probes supported by AOS: ICMP echo, TCP connect, and HTTP request. Commands common to all the probe types are identified in the following section as well as isolated commands that only apply to the specific probe types.

Additional configuration commands are available for associating tracks with each probe. These are explained in the *Network Monitor Track Configuration Command Set* on page 1698.

To activate the Network Monitor Probe Configuration mode, enter the **probe** command at the Global Configuration mode prompt followed by the probe name. Specify the probe type of **icmp-echo**, **tcp-connect**, or **http-request**. For example:

```
>enable
#configure terminal
(config)#probe probe1 icmp-echo
(config-probe-probe1)#
```

The following command is common to multiple command sets and covered in a centralized section of this guide. For more information, refer to the section listed below:

*do on page 33*
*exit on page 35*

The following commands are applicable to ICMP echo probe types and can be executed after this command:

(config)#**probe** <*probe name*> **icmp-echo**

*data <pattern> on page 1685*
*destination on page 1686*
*period <value> on page 1689*
*shutdown on page 1691*
*size <data length> on page 1692*
*source-address <ip address> on page 1693*
*timeout <value> on page 1695*
*tolerance on page 1696*

The following commands are applicable to TCP connect probe types and can be executed after this command:

(config)**#probe** <*probe name*> **tcp-connect**

*destination* *on page 1686*
*period <value>* *on page 1689*
*shutdown* *on page 1691*
*source-address <ip address>* *on page 1693*
*source-port <port>* *on page 1694*
*timeout <value>* *on page 1695*
*tolerance* *on page 1696*

The following commands are applicable to HTTP request probe types and can be executed after this command:

(config)**#probe** <*probe name*> **http-request**

*absolute-path* *on page 1684*
*destination* *on page 1686*
*expect regex <expression>* *on page 1687*
*expect status <minimum> <maximum>* *on page 1688*
*period <value>* *on page 1689*
*raw-string* *on page 1690*
*shutdown* *on page 1691*
*source-address <ip address>* *on page 1693*
*source-port <port>* *on page 1694*
*timeout <value>* *on page 1695*
*tolerance* *on page 1696*
*type* *on page 1697*

# absolute-path

Use the **absolute-path** command to specify the server's root path. Use the **no** form of this command to return to the default.

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies a pathname. |

## Default Values

By default, the path name is the forward slash symbol ( **/** ).

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Functional Notes

This command can only be executed while in the **probe** *<name>* **http-request** command set.

## Usage Examples

The following example sets the absolute-path to **/home/index.html**:

(config)**#probe probe1 http-request**
(config-probe-probe1)**#absolute-path /home/index.htm**

# data *<pattern>*

Use the **data** command to specify a hexadecimal pattern to repeat in the ICMP packet data section. Use the **no** form of this command to return to the default.

## Syntax Description

| | |
|---|---|
| *<pattern>* | Specifies a hexadecimal data pattern. |

## Default Values

By default, the data pattern is a standard ping packet pattern of data values starting with 0x00, incrementing by one for the length of the packet. Refer to *ping <ip address>* on page 20 for more information on ping packet data patterns.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Functional Notes

This command can only be executed while in the **probe** *<name>* **icmp-echo** command set.

## Usage Examples

The following example specifies a data pattern of **0FF0** for **probe1**:

```
(config)#probe probe1 icmp-echo
(config-probe-probe1)#data 0FF0
```

# destination

Use the **destination** command to specify the destination host name and port for the probe object. Use the **no** form of this command to remove the setting. Variations of this command include:

**destination** <*hostname*>
**destination** <*hostname*> **port** <*number*>
**destination** <*ip address*>
**destination** <*ip address*> **port** <*number*>

*The probe is not operational until a destination is defined.*

## Syntax Description

| | |
|---|---|
| <*hostname*> | Specifies the IP host by name. |
| <*ip address*> | Specifies a valid IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| **port** | Optional. Specifies port number. This feature is not used with icmp-echo probes. |

## Default Values

By default, there is no setting for this command.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example specifies **www.adtran.com** as the host and **port 21** (FTP) as the destination for **probe1**:

(config)**#probe probe1 http-request**
(config-probe-probe1)**#destination www.adtran.com port 21**

# expect regex *<expression>*

Use the **expect regex** command to configure the probe to expect a regular expression inside the contents of the HTTP response message. If the regular expression does not match anything, the probe fails. Use the **no** form of this command to return to the default.

## Syntax Description

| | |
|---|---|
| *<expression>* | Specifies the expression to display. |

## Default Values

By default, no regular expression is defined.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Functional Notes

This command can only be executed while in the **probe** *<name>* **http-request** command set.

## Usage Examples

The following example only allows the **probe1** test to pass if the word **success** is found in the HTTP server response message:

(config)**#probe probe1 http-request**
(config-probe-probe1)**#expect regex success**

# expect status <*minimum*> <*maximum*>

Use the **expect status** command to configure the probe to expect a specific status code in response to an HTTP request message. If a different status code is returned, the probe fails. Use the **no** form of this command to return to the default. Variations of this commands include:

**expect status** <*minimum*>
**expect status** <*minimum*> <*maximum*>

## Syntax Description

| | |
|---|---|
| <*minimum*> | Specifies a minimum number value for the status code. Valid range is 0 to 999. |
| <*maximum*> | Optional. Specifies a maximum number to create a range of status codes. Valid range is 0 to 999. |

## Default Values

By default, there is no setting for this command.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Functional Notes

This command can only be executed while in the **probe** <*name*> **http-request** command set.

Specifying only a minimum value indicates only one value can match the status code. Entering a maximum value indicates a range of possible matches.

## Usage Examples

The following example configures **probe1** to expect a status code of **200** (the status of a successful HTTP request):

(config)**#probe probe1 http-request**
(config-probe-probe1)**#expect status 200**

# period <*value*>

Use the **period** command to specify the time between probe test attempts. Use the **no** form of this command to return to the default.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the time (in seconds) between probe test attempts. Valid range is 1 to 4,294,967,295 seconds. |

## Default Values

By default, the period between probe tests is 60 seconds.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example specifies **probe1** to initiate probe tests every **90** seconds:

(config)**#probe probe1 icmp-echo**
(config-probe-probe1)**#period 90**

# raw-string

Use the **raw-string** command to enter text to appear in the data portion of an HTTP request. Refer to *ping* on page 190 for more details on the output text. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

There is no default value for this command.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Functional Notes

This command can only be executed while in the **probe** <*name*> **http-request** command set. The type should be set to RAW. Refer to *type* on page 1697 for more information.

The following system variables can be used in the text:

$SYSTEM_NAME = The host name of the system.

$SYSTEM_SERIAL_NUMBER = The serial number of the system.

$SYSTEM_DESCRIPTION = The product name and part number of the system.

$SYSTEM_SOFTWARE_VERSION = The firmware version of the system.

## Usage Examples

The following example configures a RAW HTTP request that attempts to access **update.php** on the web server. This command could be useful if the server administrator creates a PHP script which logs network connectivity information. Additional information (the router name and its uptime) placed after **update.php** is sent to the HTTP server.

```
(config)#probe probe1 http-request
(config-probe-probe1)#raw-string
    GET /update.php?hostname=$SYSTEM_NAME&uptime=$SYSTEM_UPTIME HTTP/1.0
    \r\n
    \r\n
    exit
```

# shutdown

Use the **shutdown** command to disable a probe and cause it to cease generating traffic. While a probe is shut down, it will return a fail value to a track. Use the **no** form of this command to enable a probe to generate traffic.

## Syntax Description

No subcommands.

## Default Values

By default, probes are shut down when created.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Functional Notes

A probe must be created first using the **probe** command. Refer to *probe* on page 639 for more information. Issuing the **shutdown** command at the probe configuration prompt will disable a probe, causing it to cease generating traffic.

## Usage Examples

The following example disables **probe1**:

```
(config)#probe probe1 http-request
(config-probe-probe1)#shutdown
```

## **size** *<data length>*

Use the **size** command to specify the length of the ICMP packet's data section. Use the **no** form of this command to return to the default.

### Syntax Description

| | |
|---|---|
| *<data length>* | Specifies size of ICMP datagram. Valid range is 0 to 1448 bytes. |

### Default Values

By default, the data length is 64 bytes.

### Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

### Functional Notes

This command can only be executed while in the **probe** *<name>* **icmp-echo** command set.

### Usage Examples

The following example sets the length of the ICMP packet's data section for **probe1** to 25 bytes:

```
(config)#probe probe1 icmp-echo
(config-probe-probe1)#size 25
```

# source-address <*ip address*>

Use the **source-address** command to associate an IP address source for probe traffic. Use the **no** form of this command to remove the source IP address.

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies a valid IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

## Default Values

By default, the IP address of the outbound interface is used.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Functional Notes

A valid local IP address must be entered for proper functionality.

## Usage Examples

The following example configures the source IP address on **probe1**:

(config)**#probe probe1 icmp-echo**
(config-probe-probe1)**#source-address 10.10.10.1**

# source-port <*port*>

Use the **source-port** command to specify a port source to use for probe traffic. Use the **no** form of this command to return to the default.

## Syntax Description

| | |
|---|---|
| <*port*> | Specifies the port number. Valid range is 1 to 65,535. |

## Default Values

By default, the probe automatically selects the port number.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Functional Notes

This command can be executed while in the **probe** <*name*> **tcp-connect** or **http-request** command set.

## Usage Examples

The following example configures the source port on **probe1** as **5000**:

(config)**#probe probe1 http-request**
(config-probe-probe1)**#source-port 5000**

# timeout <*value*>

Use the **timeout** command to specify the amount of time to wait for a test result before determining a failure. Use the **no** form of this command to remove the timeout setting.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the timeout value in milliseconds. This value must be less than the probe period value (refer to *period* <*value*> on page 1689). Valid range is 250 to 4,294,967,296 milliseconds. |

## Default Values

By default, the timeout is 1500 milliseconds for ICMP echo probes, 10,000 milliseconds (10 seconds) for TCP connect probes and 10,000 milliseconds (10 seconds) HTTP request probes.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example configures AOS to wait 90 milliseconds before determining a failure on **probe1**:

```
(config)#probe probe1 icmp-echo
(config-probe-probe1)#timeout 90
```

# tolerance

Use the **tolerance** command to configure the tolerance level for test failures before returning a fail status from the probe. Levels can be specified for consecutive failures or by rate of failure. Use the **no** form of this command to remove tolerance levels from probes. Variations of this command include:

**tolerance consecutive-failures** *<failures>*
**tolerance rate-of-failure** *<failures>* *<set size>*


*The probe is not operational until tolerance is defined.*

## Syntax Description

| | |
|---|---|
| **consecutive-failures** | Specifies consecutive failures to determine tolerance level. |
| **rate-of-failure** | Specifies failures-per-set to determine tolerance level. |
| *<failures>* | Specifies the number of failures before declaring the probe has failed. Valid ranges are 1 to 255 consecutive failures and 1 to 254 failures per set. |
| *<set size>* | Specifies test set size for **rate-of-failure**. Valid range is 1 to 255. |

## Default Values

By default, there are no configured tolerance levels.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example configures AOS to allow 255 consecutive failures before changing the probe status to failing:

(config)**#probe probe1 icmp-echo**
(config-probe-probe1)**#tolerance consecutive-failure 255**

The following example configures AOS to allow 254 failures per test set (where test set size is 255), before changing the probe status to failing:

(config)**#probe probe1 icmp-echo**
(config-probe-probe1)**#tolerance consecutive-failure 254 255**

# type

Use the **type** command to specify an HTTP request type. Use the **no** form of this command to return to the default. Variations of this command include:

**type get**
**type head**
**type raw**

## Syntax Description

| | |
|---|---|
| **get** | Specifies the probe use HTTP **get** request. |
| **head** | Specifies the probe use HTTP **head** request. |
| **raw** | Specifies the probe use HTTP **raw** request. |

## Default Values

By default, the probe's HTTP request is set to **get**.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Functional Notes

This command can only be executed while in the **probe** <*name*> **http-request** command set.

## Usage Examples

The following example configures **probe1** to use HTTP request **raw**:

(config)**#probe probe1 http-request**
(config-probe-probe1)**#type raw**

# NETWORK MONITOR TRACK CONFIGURATION COMMAND SET

This section explains the commands available for Network Monitoring Tracks. Tracks are objects created to monitor network probes for a change in their state. The tracks can be configured to perform a specific action based upon the probe state detected. Association between a track and a probe occurs through referencing the probe in the track's configuration. Once the track is registered with the probe, whenever a change occurs with the probe's state, an event is sent to the track.

Additional configuration commands are available for creating probes. These are explained in the *Network Monitor Probe Command Set* on page 1682.

To activate the Network Monitor Track Configuration mode, enter the **track** command at the Global Configuration mode prompt followed by the name of the track. For example:

```
>enable
#configure terminal
(config)#track track1
(config-track-track1)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:

*exit on page 35*

All other commands for this command set are described in this section in alphabetical order:



# dampening-interval *<value>*

Use the **dampening-interval** command to specify an amount of time to wait before allowing a new probe status change to trigger a new action. Use the **no** form of this command to return to the default.

## Syntax Description

| | |
|---|---|
| *<value>* | Specifies the time interval value in seconds. Valid range is 1 to 4,294,967,295. |

## Default Values

By default, the interval is set to 0 seconds.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example sets the dampening interval to 90 seconds:

(config)**#track track1**
(config-track-track1)**#dampening-interval 90**

# log-changes

Use the **log-changes** command to enable logging of status changes. When enabled, probe state transitions are displayed (real time) on the terminal (or Telnet) screen. Use the **no** form of this command to disable this feature. Unlike track debug commands, the **log-changes** command appears in the running configuration and can be saved to persist through a unit restart.

## Syntax Description

No subcommands.

## Default Values

By default, this feature is disabled.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example enables the logging of status changes:

```
(config)#track track1
(config-track-track1)#log-changes
```

# shutdown

Use the **shutdown** command to disable a track. While a track is shutdown, it is forced to fail. Use the **no** form of this command to enable a track.

## Syntax Description

No subcommands.

## Default Values

By default, tracks are active when created.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Functional Notes

A track must be created first using the **track** command in the Global Configuration mode. Refer to the command *track <name>* on page 691 for more information. Issuing the shutdown command at the track configuration prompt will force the track to fail.

## Usage Examples

The following example disables **track1**:

```
(config)#track track1
(config-track-track1)#shutdown
```

# test probe

Use the **test probe** command to base the track state upon the status of one or two probes. Two probes cans be associated with the same track by using the logical statements of AND/OR. Use the **no** form of this command to return to the default. Variations of this command include:

**test probe** <*name*>
**test probe** <*name*> **and probe** <*name*>
**test probe** <*name*> **or probe** <*name*>

## Syntax Description

| | |
|---|---|
| <*name*> | Specifies the name of the probe. |
| **and probe** | Indicates that both probes must be in the pass state for the track to pass. |
| **or probe** | Indicates that either probe can be in the pass state for the track to pass. The track only fails when both probes are in the fail state. |

## Default Values

By default, a track is not associated with any probes.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example associates the track **track1** with the probe **probe1**:

(config)**#track track1**
(config-track-track1)**#test probe probe1**

# ROUTER (OSPF) CONFIGURATION COMMAND SET

To activate the Router (OSPF) Configuration mode, enter the **router ospf** command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#router ospf
(config-ospf)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:

*cross-connect on page 29*
*do on page 33*
*end on page 34*
*exit on page 35*

All other commands for this command set are described in this section in alphabetical order.



# **area** <*area id*> **default-cost** <*value*>

Use the **area default-cost** command to assign a cost of the default summary route sent into a stub area or not-so-stubby-area (NSSA). Use the **no** form of this command to delete the assigned cost.

## Syntax Description

| | |
|---|---|
| <*area id*> | Specifies the identifier for this area. Specifies as an integer (range is 0 to 4,294,967,295) or an IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| <*value*> | Specifies the default summary route cost. Range is 0 to 166,777,214. |

## Default Values

By default the summary route cost is set to 0. There is no default for the area ID.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Usage Examples

The following example defines a default cost of 85 to a specific area:

(config)**#router ospf**
(config-ospf)**#area 192.22.72.0 default-cost 85**

# **area** *<area id>* **range** *<ip address> <subnet mask>*

Use the **area range** command to configure area route summarizations and to determine whether an address range is advertised to the networks. Use the **no** form of this command to return to disable this feature. Variations of this command include:

**area** *<area id>* **range** *<ip address>* *<subnet mask>* **advertise**
**area** *<area id>* **range** *<ip address>* *<subnet mask>* **not-advertise**

## Syntax Description

| | |
|---|---|
| *<area id>* | Specifies the identifier for this area. Specifies as an integer (range is 0 to 4,294,967,295) or an IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| *<ip address>* | Specifies the IP address of the advertised summary route. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| *<subnet mask>* | Specifies the subnet mask that corresponds to a range of IP addresses (network) or a specific host. Subnet masks can be expressed in dotted decimal notation (for example, **255.255.255.0**) or as a prefix length (for example, **/24**). |
| **advertise** | Specifies that the address range will be advertised to other networks. |
| **not-advertise** | Specifies that the address range will not be advertised to other networks. |

## Default Values

By default, OSPF is not enabled.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Usage Examples

The following example defines an address range for a specific area that allows the unit to advertise this range to other networks:

(config)**#router ospf**
(config-ospf)**#area 11.0.0.0 range 11.0.0.0 255.0.0.0 advertise**

# area <*area id*> stub

Use the **area stub** command to configure an area as a stub area. Use the **no** form of this command to disable stub-designation for areas defined as stubs using this command. Variations of this command include:

**area** <*area id*> **stub**
**area** <*area id*> **stub no-summary**

## Syntax Description

| | |
|---|---|
| *<area id>* | Specifies the identifier for this area. Specifies as an integer (range is 0 to 4,294,967,295) or an IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| **no-summary** | Optional. Designates the area as a total stub area. No summary link advertisements will be sent by the ABR into the stub area. |

## Default Values

By default, OSPF is not enabled.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Technology Review

It is important to coordinate configuration of all routers and access servers in the stub area. The **area stub** command must be configured for each of those pieces of equipment. Use the **area router configuration** command with the **area default-cost** command to specify the cost of a default internal router sent into a stub area by an ABR. Refer to *area <area id> default-cost <value>* on page 1704 for related information.

## Usage Examples

The following example configures area 2 as a stub area:

(config)**#router ospf**
(config-ospf)**#area 2 stub**

# auto-cost reference-bandwidth <*value*>

Use the **auto-cost reference-bandwidth** command to assign a different interface cost to an interface. It may be necessary to assign a higher number to high-bandwidth links. This value is used in OSPF metric calculations. Use the **no** form of this command to return to the default setting.

## Syntax Description

| | |
|---|---|
| <*value*> | Sets the default reference bandwidth rate in Mbps. Range is 1 to 4,294,967 Mbps). |

## Default Values

By default, the rate is set to 100.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Usage Examples

The following example sets the auto cost reference-bandwidth to 1000 Mbps:

(config)**#router ospf**
(config-ospf)**#auto-cost reference-bandwidth 1000**

# default-information-originate

Use the **default-information-originate** command to cause an ASBR to generate a default route. It must have its own default route before it generates one unless the **always** keyword is used. Use the **no** form of this command to return to the default setting. Variations of this command include:

**default-information-originate**
**default-information-originate always**
**default-information-originate always metric** <*value*>
**default-information-originate always metric** <*value*> **metric-type** <*type*>
**default-information-originate always metric-type** <*type*>
**default-information-originate metric** <*value*>
**default-information-originate metric** <*value*> **metric-type** <*type*>
**default-information-originate metric-type** <*type*>

## Syntax Description

| | |
|---|---|
| **always** | Optional. Specifies to always advertise default route. |
| **metric** <*value*> | Optional. Configures the metric value. Range is 0 to 16,777,214. |
| **metric type** <*type*> | Optional. Configures the metric type. Select from types 1 or 2. |

## Default Values

By default, the metric value is set to 10 and the metric type is set to 2.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Usage Examples

The following example configures a router to always advertise default routes and assigns the default router a metric value of 10000 and a metric type of 2:

(config)**#router ospf**
(config-ospf)**#default-information-originate always metric 10000 metric-type 2**

# default-metric *<value>*

Use the **default-metric** command to set a metric value for redistributed routes. Use the **no** form of this command to return to the default setting.

## Syntax Description

| | |
|---|---|
| *<value>* | Sets the default metric value. Range is 0 to 4,294,967,295. |

## Default Values

By default, **default-metric** value is set at 20.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Functional Notes

The metric value defined using the **redistribute** command overrides the **default-metric** command's metric setting. Refer to *redistribute ospf* on page 1730 for related information.

## Usage Examples

The following example shows a router using both RIP and OSPF routing protocols. The example advertises RIP-derived routes using the OSPF protocol and assigns the RIP-derived routes an OSPF metric of 10.

```
(config)#router ospf
(config-ospf)#default-metric 10
(config-ospf)#redistribute rip
```

# maximum-paths *<value>*

Use the **maximum-paths** command to specify the number of parallel routes (shared paths) OSPF can inject into the route table. When IP load sharing is enabled, traffic is balanced to a specific destination across up to six equal paths. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| *<value>* | Specifies the number of routes OSPF can insert into the route table. Valid range is 1 to 6. |

## Default Values

By default, **maximum paths** is set to **4**.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example sets the maximum number of multipath routes OSPF can insert in the route table to 5.

(config)**#router ospf**
(config-ospf)**#maximum paths 5**

# **network** *<ip address> <wildcard mask>* **area** *<area id>*

Use the **network area** command to enable routing on an IP stack and to define area IDs for the interfaces on which OSPF will run. Use the **no** form of this command to disable OSPF routing for interfaces defined using this command.

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies a valid IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| *<wildcard>* | Specifies the wildcard mask that corresponds to a range of IP addresses (network). Wildcard masks are expressed in dotted decimal notation (for example, 0.0.0.255). |
| *<area id>* | Specifies the identifier for this area. Specifies as an integer (range is 0 to 4,294,967,295) or an IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

## Default Values

No default values required for this command.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Functional Notes

In order for OSPF to operate on an interface, the *primary* address for the interface must be included in the **network area** command. Assigning an interface to an OSPF area is done using the **network area** command. There is no limit to the number of **network area** commands used on a router. If the address ranges defined for different areas overlap, the first area in the **network area** command list is used and all other overlapping portions are disregarded. Try to avoid overlapping to avoid complications.

## Usage Examples

In the following example, the OSPF routing process is enabled and two OSPF areas are defined:

(config)**#router ospf**
(config-ospf)**#network 192.22.72.101 0.0.0.255 area 0**
(config-ospf)**#network 10.0.0.0 0.255.255.255 area 10.0.0.0**

# redistribute connected

Use the **redistribute connected** command to advertise routes from one protocol to another. Using the **connected** keyword allows the advertisement of connected routes into the OSPF routing protocol. This will advertise all connected routes on OSPF-enabled interfaces. It does not enable OSPF on all interfaces. Use the **no** form of this command to disable the propagation of the specified route type. Variations of this command include:

**redistribute connected**
**redistribute connected metric** <*value*>
**redistribute connected metric-type** <*type*>
**redistribute connected subnets**

## Syntax Description

| | |
|---|---|
| **metric** <*value*> | Optional. Specifies a metric value to be carried from one OSPF process to the next (if no other value is specified). |
| **metric-type** <*type*> | Optional. Specifies a type 1 or type 2 external route as the external link type. If not specified, the default is 2. |
| **subnets** | Optional. Specifies subnet redistribution when redistributing routes into OSPF. |

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |
| Release 10.1 | Subcommands were added. |

## Functional Notes

Redistributing connected routes imports those routes into OSPF without the interfaces in question actually participating in OSPF. The connected routes imported this way are not covered by a network command and therefore do not send/receive OSPF traffic.

## Usage Examples

The following example imports connected routes into OSPF:

(config)**#router ospf**
(config-ospf)**#redistribute connected**

# redistribute

Use the **redistribute** command to advertise routes from one protocol to another, regardless of the routing protocol implemented on the routing domain. Using the **rip** keyword allows the propagation of RIP routes into OSPF. Using the **static** keyword allows the advertisement of static routes into the OSPF routing protocol. This will advertise all static routes on OSPF-enabled interfaces. It does not enable OSPF on all interfaces. Use the **no** form of this command to disable the propagation of the specified route type. Variations of this command include:

**redistribute rip**
**redistribute rip metric** <*value*>
**redistribute rip metric-type** <*type*>
**redistribute rip subnets**
**redistribute static**
**redistribute static metric** <*value*>
**redistribute static metric-type** <*type*>
**redistribute static subnets**

## Syntax Description

| | |
|---|---|
| **rip** | Specifies advertising RIP routes using OSPF. |
| **static** | Specifies advertising static routes using OSPF. |
| **metric** <*value*> | Optional. Specifies a metric value to be carried from one OSPF process to the next (if no other value is specified). |
| **metric-type** <*type*> | Optional. Specifies a type 1 or type 2 external route as the external link type. If not specified, the default is 2. |
| **subnets** | Optional. Specifies subnet redistribution when redistributing routes into OSPF. |

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |
| Release 10.1 | Subcommands were added. |

## Functional Notes

Redistributing routes imports routes (routes learned through RIP or static routes) into OSPF without the interfaces in question actually participating in OSPF. RIP and static routes imported this way are not covered by a network command and therefore do not send/receive OSPF traffic.

## Usage Examples

The following example imports RIP routes into OSPF:

(config)**#router ospf**
(config-ospf)**#redistribute rip**

The following example imports static routes into OSPF:

(config)**#router ospf**
(config-ospf)**#redistribute static**

# summary-address *<ip address> <subnet mask>*

Use the **summary-address** command to control address summarization of routes that are redistributed into OSPF from other sources (for example, RIP-to-OSPF, static-to-OSPF, etc.). Variations of this command include:

**summary address** *<ip address> <subnet mask>*
**summary address** *<ip address> <subnet mask>* **not-advertise**

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies a valid IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| *<subnet mask>* | Specifies the subnet mask that corresponds to a range of IP addresses (network) or a specific host. Subnet masks can be expressed in dotted decimal notation (for example, **255.255.255.0**) or as a prefix length (for example, **/24**). |
| **not advertise** | Optional. Causes suppression of routes that match the specified IP address and subnet mask. |

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Usage Examples

The following example suppresses advertisement of the routes which match the specified IP address and subnet mask:

(config)**#router ospf**
(config-ospf)**#summary-address 11.0.0.0 255.0.0.0 not-advertise**

# timers lsa-group-pacing *<value>*

Use the **timers lsa-group-pacing** command to change the link state advertisement (LSA) refresh interval. Use the **no** form of this command to return to the default setting.

## Syntax Description

| | |
|---|---|
| *<value>* | Sets the LSA refresh interval in seconds. Range is 10 to 1800 seconds. |

## Default Values

By default, this value is set at 240 seconds.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Usage Examples

The following example sets the refresh interval for six minutes:

(config)**#router ospf**
(config-ospf)**#timers lsa-group-pacing 360**

# timers spf *<delay> <hold>*

Use the **timers spf** command to configure the shortest path first (SPF) calculation and hold intervals. Use the **no** form of this command to return to the default setting.

## Syntax Description

| | |
|---|---|
| *<delay>* | Specifies the time in seconds between OSPF's receipt of topology changes and the beginning of SPF calculations. |
| *<hold>* | Specifies the time in seconds between consecutive SPF calculations. Range is 10 to 1800 seconds. |

## Default Values

By default, the SPF delay is 5 seconds and the hold interval is set to 10 seconds

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Usage Examples

The following example defines a delay of 10 seconds and a hold-time of 30 seconds:

(config)**#router ospf**
**(config-ospf)#**timers spf 10 30

# ROUTER (PIM SPARSE) CONFIGURATION COMMAND SET

To activate the Router (PIM Sparse) Configuration mode, enter the **router pim-sparse** command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#router pim-sparse
(config-pim-sparse)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:

*cross-connect on page 29*
*do on page 33*
*end on page 34*
*exit on page 35*

All other commands for this command set are described in this section in alphabetical order.

*join-prune-msg-interval <value> on page 1719*
*rp-address <ip address> on page 1720*
*spt-threshold on page 1721*

# join-prune-msg-interval <*value*>

Use the **join-prune-msg-interval** command to set a timing rate for PIM sparse join/prune messages. Use the **no** form of this command to return to the default setting.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the PIM sparse join/prune message interval. Valid range: 10 to 65,534 seconds. |

## Default Values

By default, the message interval is set to 60 seconds.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example sets the interval for 50 seconds:

(config)**#router pim-sparse**
(config-pim-sparse)**#join-prune-msg-interval 50**

# rp-address <*ip address*>

Use the **rp-address** command to specify a static IP address for the rendezvous point (RP) router. The **access-group** keyword is used to limit the multicast group addresses to which the RP applies. Use the **no** form of this command to remove a static IP address for the RP router. Variations of this command include:

**rp-address** <*ip address*>
**rp-address** <*ip address*> **access-group** <*name*>

## Syntax Description

| | |
|---|---|
| <*ip address*> | Specifies the IP address for the RP. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| **access-group** <*name*> | Optional. Specifies the particular access group to which the RP applies. |

## Default Values

No default necessary for this command.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Functional Notes

The **access-group** keyword is used to limit the multicast group addresses to which the RP applies. If more than one RP is configured for a given multicast group address, then a hash algorithm determines the appropriate hierarchy (as shown below). The results of the hash algorithm can be seen with the **show ip pim-sparse rp-map command**.

The hash algorithm is defined in RFC 2117 section 3.7 as follows:

For each RP address C(i) in the RP-Set, whose Group-prefix covers G, compute a value:

Value(G,M,C(i))=(1103515245 * ((1103515245 * (G&M)+12345) XOR C(i)) + 12345) mod 2^31

where M is a hash-mask included in Bootstrap messages.This hash-mask allows a small number of consecutive groups (e.g., 4) to always hash to the same RP. For instance, hierarchically-encoded data can be sent on consecutive group addresses to get the same delay and fate-sharing characteristics.

The candidate with the highest resulting value is then chosen as the RP for that group, and its identity and hash value are stored with the entry created.

Ties between C-RPs having the same hash value, are broken in advantage of the highest address.

## Usage Examples

The following example specifies an IP address of 172.22.5.100 for the RP:

(config)**#router pim-sparse**
(config-pim-sparse)**#rp-address 172.22.5.100**

# spt-threshold

Use the **spt-threshold** command to change the PIM Sparse Shortest Path Tree (SPT) threshold, which specifies the number of packets the router sends using the rendezvous point (RP) before switching to the SPT. Use the **no** form of this command to return to the default setting. Variations of this command include:

**spt-threshold** <*value*>
**spt-threshold infinity**

## Syntax Description

| | |
|---|---|
| <*value*> | Optional. Specifies the number of packets the routing switch sends using the RP before switching to the SPT. Valid range is 1 to 4,294,967,295 packets. |
| **infinity** | Optional. Causes all sources to use the shared RP tree. |

## Default Values

By default, the SPT threshold is set to 1 packet.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example sets the SPT threshold at five packets:

(config)**#router pim-sparse**
(config-pim-sparse)**#spt-threshold 5**

# ROUTER (RIP) CONFIGURATION COMMAND SET

To activate the Router (RIP) Configuration mode, enter the **router rip** command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#router rip
(config-rip)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:



All other commands for this command set are described in this section in alphabetical order.



# auto-summary

Use the **auto-summary** command to have RIP version 2 summarize subnets to the classful boundaries. Use the **no** form of this command to disable this summarization.

## Syntax Description

No subcommands.

## Default Values

By default, auto-summary is disabled.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Functional Notes

Use this command if you are subdividing a classful network into many subnets and these subnets are to be advertised over a slow link (64k or less) to a router that can only reach the classful network via the router you are configuring.

## Usage Examples

The following example configures the router to not automatically summarize network numbers:

(config)**#router rip**
(config-rip)**#no auto-summary**

# default-metric *<value>*

Use the **default-metric** command to set the default metric value for the RIP routing protocol. Use the **no** form of this command to return to the default settings.

## Syntax Description

| | |
|---|---|
| *<value>* | Sets the default metric value in Mbps. Range is 1 to 4,294,967,295 Mbps. |

## Default Values

By default, this value is set at 0.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Functional Notes

The metric value defined using the **redistribute** command overrides the **default-metric** command's metric setting. Refer to *redistribute ospf* on page 1730 for related information.

## Usage Examples

The following example shows a router using both RIP and OSPF routing protocols. The example advertises OSPF-derived routes using the RIP protocol and assigns the OSPF-derived routes a RIP metric of 10.

```
(config)#router rip
(config-rip)#default-metric 10
(config-rip)#redistribute ospf
```

# distribute-list <*name*>

Use the **distribute-list** command to add route filtering functionality by assigning inbound and outbound access control lists on either a per-interface or global basis. Only one inbound/outbound pair of access control lists can be configured for a particular interface. Use the **no** form of this command to disable the filtering. Variations of this command include:

**distribute-list** <*name*> **in**
**distribute-list** <*name*> **in** <*interface*>
**distribute-list** <*name*> **out**
**distribute-list** <*name*> **out** <*info source*>

## Syntax Description

| | |
|---|---|
| <*name*> | Specifies an access control list name. This is a standard IP access control list (ACL) against which the contents of the incoming/outgoing routing updates are matched. |
| **in** | Applies route filtering to inbound data. |
| **in** <*interface*> | Optional. Specifies the interface in which to apply the ACL. Specify an interface in the format <*interface type [slot/port | slot/port.sub-interface id | interface id | interface id.sub-interface id]*>. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **distribute-list list1 in ?** for a complete list of applicable interfaces. |
| **out** | Applies route filtering to outbound data. |
| **out** <*info source*> | Optional. Specifies the source of the routing information. The source can be an interface or a routing process (**connected**, **ospf**, **rip**, or **static**). Type **distribute list** <*name*> **out ?** for a list of available options. |

## Default Values

By default, distribute-list filtering is disabled.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example will filter out all network advertisements received via Ethernet interface 0/1 with the exception of the 10.10.10.0 network:

(config)**#router rip**
(config-rip)**#version 2**
(config-rip)**#network 192.168.1.0 255.255.255.0**
(config-rip)**#distribute-list list_1 in eth 0/1**
(config-rip)**#exit**
(config)**#ip access-list standard list_1**

(config-std-nacl)**#permit 10.10.10.0 0.0.0.255**

# **network** *<ip address> <subnet mask>*

Use the **network** command to enable RIP on the specified network. Use the **no** form of this command to remove a network from the list.

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies the IP address of the network on which RIP will be enabled. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| *<subnet mask>* | Specifies the subnet mask that corresponds to a range of IP addresses (network). Subnet masks can be expressed in dotted decimal notation (for example, **255.255.255.0**) or as a prefix length (for example, **/24**). |

## Default Values

By default, RIP is not enabled.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

The AOS will only allow processing (sending and receiving) RIP messages on interfaces with IP addresses that are contained in the networks listed using this command. All RIP messages received on interfaces not listed using this command will be discarded. To allow for receiving and participating in RIP but not for transmitting, use the passive-interface command (refer to *passive-interface <interface>* on page 1728).

## Usage Examples

The following example enables RIP on the 102.22.72.252/30, 192.45.2.0/24, and 10.200.0.0/16 networks:

(config)**#router rip**
(config-rip)**#network 102.22.72.252 255.255.255.252**
(config-rip)**#network 192.45.2.0 255.255.255.0**
(config-rip)**#network 10.200.0.0 255.255.0.0**

# passive-interface *<interface>*

Use the **passive-interface** command to disable the transmission of routing updates on the specified interface. Use the **no** form of this command to enable the transmission of routing updates on an interface.

## Syntax Description

| | |
|---|---|
| *<interface>* | Specifies an interface in the format *<interface type [slot/port \| slot/port.sub-interface id \| interface id \| interface id.sub-interface id]>*. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **passive-interface ?** for a complete list of valid interfaces. |

## Default Values

By default, RIP is not enabled.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

All routing updates received on that interface will still be processed (and advertised to other interfaces), but no updates will be transmitted to the network connected to the specified interface. Multiple **passive-interface** commands may be used to create a customized list of interfaces.

## Usage Examples

The following example disables routing updates on the Frame Relay link (labeled 1.17) and the PPP link (labeled 1):

(config)**#router rip**
(config-rip)**#passive-interface frame-relay 1.17**
(config-rip)**#passive-interface ppp 1**

# redistribute connected

Use the **redistribute connected** command to pass routes from one network to another, regardless of the routing protocol implemented on the routing domain. Using the **connected** keyword allows the propagation of routes connected to other interfaces using the RIP routing protocol. Use the **no** form of this command to disable the propagation of the specified route type. Variations of this command include:

**redistribute connected**
**redistribute connected metric** <*value*>

## Syntax Description

| | |
|---|---|
| **metric** <*value*> | Optional. Specifies the hop count to use for advertising redistributed OSPF routes in RIP. |

## Default Values

By default, RIP is not enabled.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Functional Notes

Redistributing connected routes imports those routes into RIP without the interfaces in question actually participating in RIP. The connected routes imported this way are not covered by a network command and therefore do not send/receive RIP traffic.

## Usage Examples

The following example passes the connected routes found in the route table to other networks running the RIP routing protocol:

(config)**#router rip**
(config-rip)**#redistribute connected**

# redistribute ospf

Use the **redistribute ospf** command to advertise routes from one protocol to another, regardless of the routing protocol implemented on the routing domain. Using the **ospf** keyword allows the propagation of OSPF routes into RIP. Use the **no** form of this command to disable the propagation of the specified route type. Variations of this command include:

**redistribute ospf**
**redistribute ospf metric** <*value*>

## Syntax Description

| | |
|---|---|
| **metric** <*value*> | Optional. Specifies the hop count to use for advertising redistributed OSPF routes in RIP. |

## Default Values

By default, this command is disabled.

## Command History

| | |
|---|---|
| Release 3.1 | Command was introduced. |

## Functional Notes

Redistributing OSPF routes imports those routes into RIP without the interfaces in question actually participating in RIP. The OSPF routes imported this way are not covered by a network command and therefore do not send/receive RIP traffic.

If **redistribute ospf** is enabled and no metric value is specified, the value defaults to **0**. The metric value defined using the **redistribute ospf metric** command overrides the **default-metric** command's metric setting. Refer to the section *timeout-timer* <*value*> on page 1732 for more information.

## Usage Examples

The following example imports OSPF routes into RIP:

(config)**#router rip**
(config-rip)**#redistribute ospf**

# redistribute static

Use the **redistribute static** command to pass routes from one network to another, regardless of the routing protocol implemented on the routing domain. Using the **static** keyword allows the propagation of static routes to other interfaces using the RIP routing protocol. Use the **no** form of this command to disable the propagation of the specified route type. Variations of this command include:

**redistribute static**
**redistribute static metric** <*value*>

*The gateway network for the static route must participate in RIP by using the network command for the gateway network.*

## Syntax Description

| | |
|---|---|
| **metric** <*value*> | Optional. Specifies the hop count to use for advertising redistributed OSPF routes in RIP. |

## Default Values

By default, RIP is not enabled.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Functional Notes

Redistributing static routes allows other network devices to learn about paths (not compatible with their system) without requiring manual input to each device on the network.

## Usage Examples

The following example passes the static routes found in the route table to other networks running the RIP routing protocol:

(config)**#router rip**
(config-rip)**#redistribute static**

# timeout-timer *<value>*

Use the **timeout-timer** command to set the timeout timer value for a route when it is learned via RIP. Each time a RIP update for that route is received, the timeout timer is reset to this value. If no updates for that route are received in the specified number of seconds and the timeout timer expires, the route is considered invalid, and it will be removed from the route table. Use the **no** form of this command to return to the default settings.

## Syntax Description

| | |
|---|---|
| *<value>* | Sets the timeout-timer value. Valid range is 5 to 4,294,967,295 seconds. |

## Default Values

By default, this value is set at 180 seconds.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Functional Notes

Note that the timeout timer value cannot be set to a value less than the **update-timer** value. It is recommended that this timer be set to a value that is three times the value of the **update-timer** (refer to *update-timer <value>* on page 1733).

## Usage Examples

The following example configures the router to mark routes invalid if no RIP updates for those routes are received within 120 seconds.

```
(config)#router rip
(config-rip)#timeout-timer 120
```

# update-timer <*value*>

Use the **update-timer** command to set the value of the RIP update interval timer. The RIP update interval is the number of seconds which must elapse between RIP update packet transmissions. Use the **no** form of this command to return to the default settings.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the number of seconds allowed to elapse between RIP update packet transmissions. Valid range is 5 to 4,294,967,295 seconds. |

## Default Values

By default, this value is set at 30 seconds.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Functional Notes

Note that the **timeout-timer** value cannot be set to a value less than the **update-timer** value. It is recommended that the **timeout-timer** be set to a value that is three times the value of the **update-timer**. (Refer to *timeout-timer* <*value*> on page 1732 for more information.)

## Usage Examples

The following example sets the rate at which RIP update messages are transmitted from the router to 20 seconds.

(config)**#router rip**
(config-rip)**#update-timer 20**

# version

Use the **version** command to specify (globally) the Routing Information Protocol (RIP) version used on all IP interfaces. This global configuration is overridden using the configuration commands **ip rip send version** and **ip rip receive version**. Use the **no** form of this command to return to the default value. Variations of this command include:

**version 1**
**version 2**

## Syntax Description

| | |
|---|---|
| **1** | Specifies RIP version 1 be used globally. |
| **2** | Specifies RIP version 2 be used globally. |

## Default Values

By default, RIP is not enabled.

## Command History

| | |
|---|---|
| Release 1.1 | Command was introduced. |

## Usage Examples

The following example specifies RIP version 2 as the global RIP version:

(config)**#router rip**
(config-rip)**#version 2**

# ISDN GROUP CONFIGURATION COMMAND SET

To activate the ISDN Group Configuration mode, enter the **isdn-group** command at the Global configuration command prompt. For example:

```
>enable
#configure terminal
(config)#isdn-group 1
(config-isdn-group 1)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:

*cross-connect* *on page 29*
*do* *on page 33*
*exit* *on page 35*

All other commands for this command set are described in this section in alphabetical order.



# accept-number <*number*>

Use the **accept-number** command to specify the incoming number to be accepted by this ISDN group. This number will be accepted and passed from the Network to the end users.

## Syntax Description

| | |
|---|---|
| <*number*> | Specifies the incoming number to be accepted by this ISDN group. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example specifies that ISDN group **1** will accept calls from 256-555-1234:

(config)**#isdn-group 1**
(config-isdn-group 1)**#accept-number 2565551234**

# call-type voice

Use the **call-type** command to specify the type of communication allocated for this group . Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| **voice** | Specifies use as voiceband line. |

## Default Values

By default, the call type is set to **voice**.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example sets the call type for ISDN group **1** to **voice**:

(config)**#isdn-group 1**
(config-isdn-group 1)**#call-type voice**

# connect pri

Use the **connect pri** command to associate a specific interface with the ISDN group. Use the **no** form of this command to disconnect the specified interface from the ISDN group.

## Syntax Description

| | |
|---|---|
| **pri** | Connects a PRI interface to the ISDN group. Type **connect ?** for a list of valid interfaces. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example associates the **pri 1** interface with ISDN group **1**:

(config)**#isdn-group 1**
(config-isdn-group 1)**#connect pri 1**

# incoming-accept-number <*number*>

Use the **incoming-accept-number** command to configure the incoming number to be accepted by this group from the PSTN. Use the **no** form of this command to remove a configured accept number.

## Syntax Description

| | |
|---|---|
| <*number*> | Specifies the phone number(s) accepted for this ISDN group. The accept number entered should match the digits that populate the Called Party Information Element received on the ISDN interface for the call. Refer to the *Functional Notes* for more information on entering the number. |

## Default Values

By default, there are no configured incoming accept numbers. The ISDN group will not be able to accept calls without a configured incoming accept number.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Functional Notes

Special characters (parentheses, commas, and dashes) can be entered in the incoming-accept-number for readability, but they are ignored by the system. Incoming-accept-numbers are entered as a single number, or as a range of numbers using the available wildcard characters. The following wildcards can be used to define numbers:

| | |
|---|---|
| **X** | Any single digit 0 through 9 |
| **N** | Any single digit 2 through 9 |
| **[1,2,3]** | Specifies single digit in this group |
| **$** | Any number; effectively functions as a “don’t care” |

The following list provides some examples for proper wildcard usage:

| **Incoming Accept Number(s)** | **Entry** |
|---|---|
| 555-1111 and 555-1112 | 555-111[1,2] |
| All numbers from the 916 area code | 916$ |
| Numbers between 555-1000 and 555-2000 | 555-[1,2]XXX |

Wildcard characters are especially useful in situations where ISDN hunt groups are deployed and the ISDN interfaces are all assigned to the same ISDN group in the router. ISDN hunt groups “bundle” multiple ISDN interfaces (with unique LDNs) together into a single group at the central office. When a call to any of the LDNs assigned to the ISDN interfaces in the hunt group is received at the central office, the switch sends the call to the first available ISDN interface. The ISDN group must be able to accept calls to multiple LDNs. Wildcard characters can simplify a configuration by allowing a single entry to match several numbers.

## Usage Examples

The following example configures the group to accept calls for 256-555-1000 through 256-555-2000:

(config)**#isdn-group 1**
(config-isdn-group 1)**#incoming-accept-number 256-555-[1,2]XXX**

# max-channels <*value*>

Use the **max-channels** command to specify the maximum number of channels allocated for the ISDN group. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the maximum number of channels allocated for the ISDN group. Valid range is from 1 to 255 channels. |

## Default Values

By default, the maximum number of channels is set to 0. When **max-channels** is set to 0, the group does not limit the number of usable channels and can use all available channels. Use the **no max-channels** command to return to the default value.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example sets the maximum number of channels for ISDN group **1** to **50**:

(config)**#isdn-group 1**
(config-isdn-group 1)**#max-channels 50**

# min-channels *<value>*

Use the **min-channels** command to specify the minimum number of channels allocated for the ISDN group. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| *<value>* | Specifies the minimum number of channels allocated for the ISDN group. Valid range is from 1 to 255 channels. |

## Default Values

By default, the minimum number of channels is set to 0. When **min-channels** is set to 0, no channels are reserved for this group. This group can use available channels, but does not have any channels specifically reserved. Use the **no min-channels** command to return to the default value.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example sets the minimum number of channels for ISDN group **1** to **10**:

(config)**#isdn-group 1**
(config-isdn-group 1)**#min-channels 10**

# resource pool-member <*name*>

Use the **resource pool-member** command to assign the group to a resource pool, making it a demand routing resource. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*name*> | Specifies the name of the resource pool to which this group is assigned. |

## Default Values

By default, the group is not assigned to any resource pool.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example configures the ISDN group 1 as a member of resource pool **MyPool**:

(config)**#isdn-group 1**
(config-isdn-group 1)**#resource pool-member** MyPool

# VOICE AUTO ATTENDANT COMMAND SET

To activate the Voice Auto Attendant Configuration mode, enter the **voice autoattendant** <*name*> <*extension*> command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#voice autoattendant Example 1212
(config-aa1212)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:



All other commands for this command set are described in this section in alphabetical order.



## entry-filename *<name>*

Use the **entry-filename** command to enter the XML file to use for this auto attendant. Use the **no** form of the command to disable the settings.

### Syntax Description

| | |
|---|---|
| *<name>* | Specifies the name of the XML filename to use for this auto attendant. |

### Default Values

No defaults necessary for this command.

### Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

### Usage Examples

The following example shows the **entry-filename** command being executed to select the XML file to use for this auto attendant:

(config)**#voice autoattendant Example 1212**
(config-aa1212)**#entry-filename Operaa**

# sip-identity

Use the **sip-identity** command to configure SIP registration options for the user. Use the **no** form of the command to disable the settings. Variations of this command include the following:

**sip-identity** <*station*> <*Txx*>
**sip-identity** <*station*> <*Txx*> **register**
**sip-identity** <*station*> <*Txx*> **register auth-name** <*username*> **password** <*password*>

## Syntax Description

| | |
|---|---|
| <*station*> <*Txx*> | Specifies the station to be used for SIP trunk (e.g., station extension) and the SIP trunk in the format Txx, (e.g., T01) through which you will register to the server. |
| **register** | Registers the user to the server. |
| **register auth-name** <*username*> | Sets the user name that will be required as authentication for registration to the SIP server. |
| **password** <*password*> | Sets the password that will be required as authentication for registration to the SIP server. |

## Default Values

No default values necessary for this command.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example configures the autoattendant to use extension **5000** as its identity on trunk **T02**:

(config)**#voice autoattendant Example 1212**
(config-aa1212)**#sip-identity 5000 T02**

# VOICE CODEC LIST CONFIGURATION COMMAND SET

To activate the Voice CODEC List Configuration mode, enter the **voice codec-list trunk** command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#voice codec-list trunk
(config-codec)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:

*cross-connect on page 29*
*description <text> on page 32*
*do on page 33*
*exit on page 35*

All other commands for this command set are described in this section in alphabetical order.

*codec on page 1748*
*default on page 1750*

# codec

Use the **codec** command to select either the g711ulaw or g729 CODEC for call negotiation for a specific CODEC list. For information on creating CODEC lists for call negotiation, see *codec-group <name>* on page 1927. Use the **no** form of this command to remove a CODEC list. Variations of this command include:

**codec g711ulaw**
**codec g729**

## Syntax Description

| | |
|---|---|
| **g711ulaw** | Assigns **g711ulaw** as the preferred CODEC for negotiation. |
| **g729** | Assigns **g729** (support for Annex A - no VAD) as the preferred CODEC for negotiation. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Functional Notes

The primary reason to assign CODEC lists is to save time. The lists can be created once and applied to many users. CODEC lists are lists of CODECs arranged in preferred order with the first listed CODEC being the most preferred for negotiation. The order of preference is used primarily to conserve bandwidth on WAN-based interfaces.

For example, create a CODEC list for users where g711ulaw is the first and g729 is the second CODEC listed (see *Usage Examples* on the next page). Create a second list for trunks, where g729 is listed first and g711ulaw is listed second. Apply the user CODEC list to the users of interest, and apply the trunk CODEC list to the trunks of interest.

When a user makes an outbound call from an FXS port to a SIP trunk, the trunk will look at its CODEC list and query the user making the call as to which CODEC is to be used according to its CODEC list. It will query with the first CODEC (which in this example is g729). If this CODEC is listed in the CODEC list that is applied to the user (even though it is second), then g729 is agreed upon and the call will be converted and sent out the trunk. This is the most bandwidth conservative CODEC in this case.

When a user makes an outbound call from an FXS port to a SIP trunk, the trunk will query the user with the first CODEC on its list (which in this example is g729). If g729 is listed in the user's CODEC list (even though it is second), then g729 is agreed upon and the call will be converted and sent out the trunk. This is the most bandwidth conservative CODEC in this case.

Now imagine another analog user, in this case a fax machine where g729 is not viable. The user's CODEC list will only include g711ulaw. (You can use the **codec** command in the user command set and assign g711ulaw directly to the user rather than creating a separate list for this task. It should be noted that if an individual CODEC is assigned using the **codec** command and a CODEC list is also defined, then the individual CODEC will be processed first and the CODEC list will be processed second.) When the fax machine makes an outbound call, the trunk will again query the user if g729 is available. This query will fail to negotiate, and the trunk will query the user with the next CODEC in its list (g711ulaw). This CODEC will match the one listed for the user and be used to send the fax out the trunk.

## Usage Examples

The following example selects the g729a CODEC (support for Annex A - no VAD) for call negotiation for the CODEC list **trunk**.

(config)**#voice codec-list trunk**
(config-codec)**#codec g729a**

# default

Use the **default** command to set a CODEC list as the default for call negotiation. Use the **no** form of this command to remove a default CODEC list.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example sets the CODEC list **trunk** as the default for call negotiation:

(config)**#voice codec-list trunk**
(config-codec)**#default**

# VOICE COS COMMAND SET

To activate the Voice Class-of-Service (CoS) Configuration mode, enter the **voice class-of-service** command at the Global Configuration command prompt. For example:

```
>enable
#configure terminal
(config)#voice class-of-service set1
Configuring Existing Level “set1”.
(config-cos-set1)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:



All other commands for this command set are described in this section in alphabetical order.



# aa-initiate-permit *<template>*

Use the **aa-initiate-permit** command to enable the hands free auto-answer feature. Hands free voice communication (similar to using a speakerphone or intercom) will be available for the programmed number template. When using **aa-initiate-permit**, the receiving party's phone automatically answers the phone. Use the **no** form of this command to disable this feature.

## Syntax Description

| | |
|---|---|
| *<template>* | Allows users with the specified number template to initiate auto-answer calls. |

## Default Values

By default, **aa-initiate-permit** is disabled.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Functional Notes

In order to place and Auto-Answer call you must dial ****** prior to the number. User may also program ****** as a softkey to dial ****** separately before dialing the number to call. Users must dial ***971** to block Auto-Answer calls and ***970** to reactivate the feature.

## Usage Examples

The following example enables the hands free auto-answer for users with the extension range of 4200 to 4299:

(config)**#voice class-of-service**
(config-T01)**#aa-initiate-permit 42xx**

# billing-codes

Use the **billing-codes** command to enable a billing account code collection. If enabled, users must enter a billing code prior to dialing a number. This feature is useful for controlling access the to long distance privileges. Use the **no** form of this command to disable account code collection.

## Syntax Description

No subcommands.

## Default Values

By default, billing codes are disabled.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example enables account code collection in rule set **set1**:

(config)**#voice class-of-service set1**
Configuring Existing Level “set1”.
(config-cos-set1)**#billing-codes**

# block-caller-id

Use the **block-caller-id** command to conceal caller ID information for outbound calls. Use the **no** form of this command to allow caller ID for outbound calls.

## Syntax Description

No subcommands.

## Default Values

By default, the **block-callerid** feature is disabled.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example blocks caller ID for outbound calls in rule set **set1**:

(config)**#voice class-of-service set1**
Configuring Existing Level “set1”.
(config-cos-set1)**#block-callerid**

# call-privilege

Use the **call-privilege** command to assign general call privileges for outbound access. This determines what type of calls a user is permitted to make as a member of this CoS. Use the **no** form of this command to remove general call privileges for outbound access. Variations of this command include:

**call-privilege 900-number**
**call-privilege all**
**call-privilege extensions**
**call-privilege international**
**call-privilege local**
**call-privilege long-distance**
**call-privilege operator-assisted**
**call-privilege specify-carrier**
**call-privilege toll-free**
**call-privilege [user1 | user2 | user3]**

## Syntax Description

| | |
|---|---|
| **900-number** | Permits 900 calls in the form 1-900-NXX-XXXX and 976-XXXX. |
| **all** | Permits all calls. |
| **extensions** | Permits internal calls. |
| **international** | Permits international calls in the form 011-number. |
| **local** | Permits local calls in the form NXX-XXXX. |
| **long-distance** | Permits long distance calls in the form 1-NXX-NXX-XXXX. |
| **operator-assisted** | Permits operator assisted calls. |
| **specify-carrier** | Permits calls that specify carrier. |
| **toll-free** | Permits toll free calls. |
| **user1** | Permits calls that match the first user-defined template. |
| **user2** | Permits calls that match the second user-defined template. |
| **user3** | Permits calls that match the third user-defined template. |
| Valid characters include: | 0-9 - Any single digit. |
| | X - Any single digit 0 through 9. |
| | N - Any single digit 2 through 9. |

## Default Values

By default, no call privileges are enabled.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example permits long distance calls in rule set **set1**:

(config)**#voice class-of-service set1**
Configuring Existing Level “set1”.
(config-cos-set1)**#call-privilege long-distance**

# camp-on

Use the **camp-on** command to allow automatic retry of a busy extension. This feature enables a user to reach the busy party as soon as the line is available. Use the **no** form of this command to disable automatic retry.

## Syntax Description

No subcommands.

## Default Values

By default, the **camp-on** feature is disabled.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example enables automatic retry of a busy extension in rule set **set1**:

```
(config)#voice class-of-service set1
Configuring Existing Level “set1”.
(config-cos-set1)#camp-on
```

# conference

Use the **conference** command to allow initiation of three-way conference calls. This feature allows multiple parties to communicate at the same time on the same line. Use the **no** form of this command to return to the default setting.

## Syntax Description

No subcommands.

## Default Values

By default, three-way conference call feature is disabled.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example allows the initiation of three-way conference calls in rule set **set1**:

(config)**#voice class-of-service set1**
Configuring Existing Level “set1”.
(config-cos-set1)**#conference**

# default-level

Use the **default-level** command to set this CoS level as the default. When enabled, new users that are added to the system are assigned this CoS by default. To change the default from this CoS level, use the **no** form of this command.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example sets the rule set **set1** as the default CoS level:

(config)**#voice class-of-service set1**
Configuring Existing Level “set1”.
(config-cos-set1)**#default-level**

# deny-template *<template>*

Use the **deny-template** command to configure a number template that specifies the types of calls that users in this CoS are not allowed to make. Use the **no** form of this command to remove a deny template.

## Syntax Descrip*tion*

| | |
|---|---|
| *<template>* | Specifies a number template. All calls matching this pattern will be denied. Valid characters include:<br>0-9 - Any single digit.<br>X - Any single digit 0 through 9.<br>N - Any single digit 2 through 9.<br>M - Any single digit 2 through 8.<br>[] - Any single digit of those within the brackets.<br>-(), - Punctuation characters that are ignored.<br>For example: 555-81XX matches 555-8100 through 555-8199 and 555-812[0,1,2] matches 555-8120 through 555-8122. |

## Default Values

No default necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example denies users in rule set **set1** the ability to make any call beginning with **555**:

(config)**#voice class-of-service set1**
Configuring Existing Level “set1”.
(config-cos-set1)**#deny-template 555-xxxx**

# disable-callwaiting

Use the **disable-callwaiting** command to allow users to control the call waiting feature. Disabling call waiting will block the alert of a incoming call while the user in on the phone. Use the **no** form of this command to enable call waiting.

## Syntax Description

No subcommands.

## Default Values

By default, call waiting is enabled.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example allows users in rule set **set1** to disable call waiting:

(config)**#voice class-of-service set1**
Configuring Existing Level “set1”.
(config-cos-set1)**#disable-callwaiting**

# dnd

Use the **dnd** command to enable do-not-disturb. Do-not-disturb makes the line appear busy to incoming calls. Use the **no** form of this command return to the default setting.

## Syntax Description

No subcommands.

## Default Values

By default, the do-not-disturb feature is disabled.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example enables the do-not-disturb feature in rule set **set1**:

```
(config)#voice class-of-service set1
Configuring Existing Level “set1”.
(config-cos-set1)#dnd
```

# door-phone

Use the **door-phone** command to allow the user to call the door phone using a special prefix (SPRE) code. Use the **no** form of this command to deny door phone access.

## Syntax Description

No subcommands.

## Default Values

By default, door-phone access is denied.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example allows door phone access in voice class-of-service rule set named **set1**:

(config)**#voice class-of-service set1**
Configuring Existing Level “set1”.
(config-cos-set1)**#door-phone**

# external-fwd

Use the **external-fwd** command to allow forwarding of calls to an external number. When enabled, the users can forward their phones to lines outside the system such as their home numbers. Use the **no** form of this command to disable external call forwarding.

## Syntax Description

No subcommands.

## Default Values

By default, forwarding calls to an external number is disabled.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example allows users in rule set **set1** to forward calls to an external number:

(config)**#voice class-of-service set1**
Configuring Existing Level “set1”.
(config-cos-set1)**#external-fwd**

# forward

Use the **forward** command to allow users to forward calls to another extension. Forwarding calls allows the use to receive incoming calls at a different number. Use the **no** form of this command to end call forwarding.

## Syntax Description

No subcommands.

## Default Values

By default, internal call forwarding is disabled.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example allows users in rule set **set1** to forward calls:

(config)**#voice class-of-service set1**
Configuring Existing Level “set1”.
(config-cos-set1)**#forward**

# hold

Use the **hold** command to allow users to place calls on standby. Use the **no** form of this command to disable the call hold option.

## Syntax Description

No subcommands.

## Default Values

By default, the call hold option is disabled.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example enables users in rule set **set1** to place a call on hold:

(config)**#voice class-of-service set1**
Configuring Existing Level “set1”.
(config-cos-set1)**#hold**

# hotel

Use the **hotel** command to allow extension reassignment to an alternate phone. Use the **no** form of this command to disable extension reassignment.

## Syntax Description

No subcommands.

## Default Values

By default, the **hotel** feature is disabled.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example enables users in rule set **set1** to reassign an extension to an alternate phone:

(config)**#voice class-of-service set1**
Configuring Existing Level “set1”.
(config-cos-set1)**#hotel**

# logout-group

Use the **logout-group** command to allow a user to issue a special prefix (SPRE) command to log out of a user group. Use the **no** form of this command to deny the ability to log out of a user group.

## Syntax Description

No subcommands.

## Default Values

By default, the ability to log out of a user group is denied.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example allows the user in rule set **set1** to log out of a user group:

```
(config)#voice class-of-service set1
Configuring Existing Level “set1”.
(config-cos-set1)#logout-group
```

# message-waiting

Use the **message-waiting** command to allow message waiting indicator control. This allows users to change the manner in which message notification takes place. Use the **no** form of this command to disable message waiting indicator control.

## Syntax Description

No subcommands.

## Default Values

By default, control of the message waiting indicator is disabled.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example allows users in rule set **set1** to manage message waiting indicators:

(config)**#voice class-of-service set1**
Configuring Existing Level “set1”.
(config-cos-set1)**#message-waiting**

# overhead-paging

Use the **overhead-paging** command to allow the user to connect to overhead paging using a special prefix (SPRE) code. Use the **no** form of this command to deny the ability to connect to overhead paging.

## Syntax Description

No subcommands.

## Default Values

By default, the ability to connect to overhead paging is denied.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example allows the user in rule set **set1** to connect to overhead paging:

(config)**#voice class-of-service set1**
Configuring Existing Level “set1”.
(config-cos-set1)**#overhead-paging**

# override-passcode *<passcode>*

Use the **override-passcode** command to assign an override passcode. This four-digit code is used in conjunction with the CoS override feature and enables a user to override an extension's configured CoS with the new CoS as defined by the passcode. Use the **no** form of this command to remove an override passcode.

## Syntax Description

| | |
|---|---|
| *<passcode>* | Specifies a four-digit numerical passcode. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example sets the override passcode to **1234** in rule set **set1**:

(config)**#voice class-of-service set1**
Configuring Existing Level "set1".
(config-cos-set1)**#override-passcode 1234**

# park

Use the **park** command to allow users to park calls on the system. Use the **no** form of this command to disable the call parking feature.

## Syntax Description

No subcommands.

## Default Values

By default, the parking feature is enabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example allows users to park calls in rule set **set1**:

(config)**#voice class-of-service set1**
Configuring Existing Level “set1”.
(config-cos-set1)**#park**

# permit template <*number*>

Use the **permit template** command to configure call privilege additions that are permitted access only. This specifically allows a otherwise restricted number to be dialed. Use the **no** form of this command to remove the template.

## Syntax Description

| | |
|---|---|
| <*number*> | Specifies the number that may be dialed. The number is in the form NXX-XXXX, where N is 2 to 9 and X is 0 to 9. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example allows the number **325-1234** to be dialed in rule set **set1**:

(config)**#voice class-of-service set1**
Configuring Existing Level “set1”.
(config-cos-set1)**#permit template 325-1234**

# program-user-speed

Use the **program-user-speed** command to allow users to access speed dial functionality through the system. Use the **no** form of this command to disable the speed dialing feature.

## Syntax Description

No subcommands.

## Default Values

By default, the speed dial feature is disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example enables the speed dial feature for users in rule set **set1**:

(config)**#voice class-of-service set1**
Configuring Existing Level “set1”.
(config-cos-set1)**#program-user-speed**

# redial

Use the **redial** command to grant users access to the redial functionality which redials the last outgoing number. Use the **no** form of this command to disable last number redial.

## Syntax Description

No subcommands.

## Default Values

By default, the last number redial feature is disabled.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example enables the last number redial feature in rule set **set1**:

(config)**#voice class-of-service set1**
Configuring Existing Level “set1”.
(config-cos-set1)**#redial**

# remote-fwd

Use the **remote-fwd** command to allow a user to control call forwarding from a remote location. Use the **no** form of this command to disable remote forwarding.

## Syntax Description

No subcommands.

## Default Values

By default, remote forwarding feature is disabled.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example allows users in rule set **set1** to enable call forwarding from a remote location:

(config)**#voice class-of-service set1**
Configuring Existing Level “set1”.
(config-cos-set1)**#remote-fwd**

# **rename** *<name>*

Use the **rename** command to assign a new name to the CoS rule set.

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies the new name of the CoS rule set. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example changed the name of rule set **set1** to **accounting**:

(config)**#voice class-of-service set1**
Configuring Existing Level “set1”.
(config-cos-set1)**#rename accounting**
(config-cos-accounting)**#**

# retrieve-park

Use the **retrieve-park** command to allow the retrieval of parked calls. Use the **no** form of this command to disable the retrieve parked calls feature.

## Syntax Description

No subcommands.

## Default Values

By default, the retrieve parked calls feature is disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example enables retrieval of parked calls in rule set **set1**:

(config)**#voice class-of-service set1**
Configuring Existing Level “set1”.
(config-cos-set1)**#retrieve-park**

# return-last-call

Use the **return-last-call** command to allow users to return the last call received. Use the **no** form of this command to disable the return last call received feature.

## Syntax Description

No subcommands.

## Default Values

By default, the **return-last-call** feature is disabled.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example allows users in rule set **set1** to return the last call received:

(config)**#voice class-of-service set1**
Configuring Existing Level “set1”.
(config-cos-set1)**#return-last-call**

# station-lock

Use the **station-lock** command to allow users to lock an extension, preventing it from making outbound calls. Use the **no** form of this command to revoke user's ability to disable outbound calling capabilities.

## Syntax Description

No subcommands.

## Default Values

By default, the **station-lock** feature is disabled. Users are not allowed to block their extension's capability to place outbound calls.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example enables the station lock feature in rule set **set1**:

(config)**#voice class-of-service set1**
Configuring Existing Level "set1".
(config-cos-set1)**#station-lock**

# system-speed

Use the **system-speed** command to enable system speed dial usage for the system. Use the **no** form of this command to disable the system speed setting.

## Syntax Description

No subcommands.

## Default Values

By default, **system-speed** is enabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example enables configuration of the system speed in rule set **set1**:

```
(config)#voice class-of-service set1
Configuring Existing Level “set1”.
(config-cos-set1)#system-speed
```

# transfer

Use the **transfer** command to allow users to perform call transfers. Use the **no** form of this command to disable call transfers.

## Syntax Description

No subcommands.

## Default Values

By default, the call transfer feature is disabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example allows users in rule set **set1** to perform call transfers:

(config)**#voice class-of-service set1**
Configuring Existing Level “set1”.
(config-cos-set1)**#transfer**

# unlock-door

Use the **unlock-door** command to enable the user to use a special prefix (SPRE) code to control the door contact. Use the **no** form of this command to disable door contact operation.

## Syntax Description

No subcommands.

## Default Values

By default, door contact operation is disabled.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example enables door contact operation in rule set **set1**:

(config)**#voice class-of-service set1**
Configuring Existing Level “set1”.
(config-cos-set1)**#unlock-door**

# user-speed

Use the **user-speed** command to allow users to program speed dial numbers. Use the **no** form of this command to deny this privilege.

## Syntax Description

No subcommands.

## Default Values

By default, user speed dial programming is not allowed.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example allows users in rule set **set1** to program speed dial numbers:

(config)**#voice class-of-service set1**
Configuring Existing Level “set1”.
(config-cos-set1)**#user-speed**

# VOICE MAIL COS COMMAND SET

To activate the Voice Mail Class-of-Service (CoS) Configuration mode, enter the **voice mail class-of-service** command at the Global Configuration command prompt. For example:

```
>enable
#configure terminal
(config)#voice mail class-of-service class1
Configuring Existing Level “class1”.
(config-vm-class1)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:

*do* *on page 33*
*exit* *on page 35*

All other commands for this command set are described in this section in alphabetical order.



# expire-time *<days>*

Use the **expire-time** command to set the number of days before a voice mail message expires. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| *<days>* | Specifies the number of days until a message expires. Valid range is 5 to 60 days. A value of 0 means messages will never expire. |

## Default Values

By default, the number of days is set to 0 which means messages will never expire.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example configures the voice mail CoS **class1** to delete voice mail messages after **14** days:

(config)**#voice mail class-of-service class1**
Configuring Existing Level "class1".
(config-vm-class1)**#expire-time 14**

# greeting-length-max *<time>*

Use the **greeting-length-max** command to set the maximum length (in seconds) for a voice mail greeting. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| *<time>* | Specifies the length in seconds for a voice mail greeting. Valid range is 20 to 120 seconds. |

## Default Values

By default, the maximum voice mail greeting time is set to 60 seconds.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example configures the maximum length for a voice mail greeting in rule set **class1** to **60** seconds:

(config)**#voice mail class-of-service class1**
Configuring Existing Level "class1".
(config-vm-class1)**#greeting-length-max 60**

# greeting-quota *<time>*

Use the **greeting-quota** command to set the maximum storage time (in minutes) of all greeting messages. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| *<time>* | Specifies the maximum storage time (in minutes) for the storage of all greeting messages. Valid range is 1 to 9 minutes. |

## Default Values

By default, the maximum storage time for all greeting messages is 3 minutes.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example configures the maximum storage time for all greeting messages in rule set **class1** to **5** minutes:

(config)**#voice mail class-of-service class1**
Configuring Existing Level "class1".
(config-vm-class1)**#greeting-quota 5**

# message-length-max *<time>*

Use the **message-length-max** command to set the maximum length (in seconds) for a voice mail message. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| *<time>* | Specifies the maximum length (in seconds) for a voice mail message. Valid range is 30 to 600 seconds. |

## Default Values

By default, the maximum length for a voice mail message is 120 seconds.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example sets the maximum length a voice mail message in rule set **class1** to **300** seconds:

(config)**#voice mail class-of-service class1**
Configuring Existing Level "class1".
(config-vm-class1)**#message-length-max 300**

# message-quota *<time>*

Use the **message-quota** command to set the maximum storage time (in minutes) of all voice mail messages. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| *<time>* | Specifies the maximum storage time (in minutes) of all voice mail messages. Valid range is 1 to 180 minutes. |

## Default Values

By default, the maximum storage time of all voice mail messages is 10 minutes.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example sets the maximum storage time of all voice mail messages in rule set **class1** to **120** minutes:

(config)**#voice mail class-of-service class1**
Configuring Existing Level "class1".
(config-vm-class1)**#message-quota 120**

# prompt-delete

Use the **prompt-delete** command to configure to unit to prompt the user before deleting messages. Use the **no** form of this command to disable the prompt before deleting messages.

## Syntax Description

No subcommands.

## Default Values

By default, the prompt before deleting messages is disabled.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example configures the unit to prompt the user before deleting voice mail messages:

(config)**#voice mail class-of-service class1**
Configuring Existing Level "class1".
(config-vm-class1)**#prompt-delete**

# rename *<name>*

Use the **rename** command to rename the voice mail class-of-service (CoS) rule set. Use the **no** form of this command to return to the previous name.

## Syntax Description<

| | |
|---|---|
| *<name>* | Specifies the new name of the CoS rule set. |

## Default Values

No default necessary for this command.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example assigns a new name (**class2)** to the current CoS rule set **class1**:

(config)**#voice mail class-of-service class1**
Configuring Existing Level "class1".
(config-vm-class1)**#rename class2**

# VOICE MAIL NOTIFY SCHEDULE COMMAND SET

To activate the Voice Mail Notify Schedule configuration mode, enter the **voice mail schedule** command at the Global Configuration command prompt. For example:

```
>enable
#configure terminal
(config)#voice user 4444
(config-4444)#voicemail notify schedule monday 06:00 am
Configuring New Schedule "monday 06:00 am".
(config-4444-mon-06:00am)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:

*do* *on page 33*
*exit* *on page 35*

All other commands for this command set are described in this section in alphabetical order.

*notify email* *on page 1795*
*show voice mail notify-schedule* *on page 1796*

# notify email

Use the **notify email** command to configure voice mail notifications for this extension. Use the **no** form of this command to delete the voice mail notifications. Variations of this command include:

**notify email primary**
**notify email secondary**

## Syntax Description

| | |
|---|---|
| **primary** | Specifies that email notifications for this schedule will be sent to the primary email address. This email address is configured using the **email** command. Refer to *email <address>* on page 1936. |
| **secondary** | Specifies that email notifications for this schedule will be sent to the secondary email address. This email address is configured with the **email-secondary** command. Refer to *email-secondary <address>* on page 1937. |

## Default Values

By default, the notifications are sent to the primary email address.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example configures the unit to send email notification for this schedule to the secondary email address.

(config)**#voice user 4444**
(config-4444)**#voicemail notify schedule monday 06:00 am**
Configuring New Schedule "monday 06:00 am".
(config-4444-mon-06:00am)**#notify email secondary**

# show voice mail notify-schedule

Use the **show voice mail notify-schedule** command to the display voice mail notification schedule.

## Syntax Description

No subcommands.

## Default Values

No default necessary for this command.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following is sample output for the **show voice mail notify-schedule** command:

```
(config)#voice user 4444
(config-4444)#voicemail notify schedule monday 06:00 am
Configuring New Schedule "monday 06:00 am".
(config-4444-mon-06:00am)#show voice mail notify-schedule

Start                End             Email 1         Email 2
-------------------------------------------------------------------------------------
Sun 12:00am      Sun 11:59pm         ---             ---
Mon 12:00am      Mon 5:59am          ---             ---
Mon  6:00am       at 11:59pm         ---             ---
```

# VOICE OPERATOR GROUP COMMAND SET

To activate the Voice Operator Group Interface Configuration mode, enter the **voice operator-group** command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#voice operator-group
(config-operator-group)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:



All other commands for this command set are described in this section in alphabetical order.



# coverage

Use the **coverage** command to configure call coverage parameters for the user. The setting of this command tells the system how to handle calls after a certain number of rings go unanswered. Use the **no** form of this command to disable coverage. Variations of this command include:

**coverage aa**
**coverage aa** <*number*>
**coverage external** <*number*>
**coverage external** <*number*> **num-ring** <*rings*>
**coverage internal** <*number*>
**coverage internal** <*number*> **num-ring** <*rings*>
**coverage vm**

## Syntax Description

| | |
|---|---|
| **aa** <*number*> | Forwards the phone to the auto attendant. It is optional to enter a specific extension programmed to forward. |
| **external** <*number*> | Forwards the phone to an external number. |
| **internal** <*number*> | Forwards the phone to an internal number. |
| **num-ring** <*rings*> | Optional. Select the number of rings for the external number before performing the next action. Valid range is 1 to 9 rings. |
| **vm** | Forwards the phone to voicemail. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |
| Release 11.1 | Command was updated to include the number of rings option. |
| Release 12.1 | Command was updated to include the auto attendant option. |
| Release 13.1 | Command was updated to include the internal and voicemail options. |

## Usage Examples

The following example specifies that the user's phone be forwarded to the internal extension **8500** after **3** rings.

(config)**#voice ring-group 1234**
(config-1234)**#coverage internal 8500 num-ring 3**

# **did** <*number*>

Use the **did** command to configure Direct Inward Dialing (DID) for this group. DID is used if a service provider is providing digits to the unit on inbound calls or if the unit needs to provide DID information to a piece of CPE equipment. Use the **no** form of this command to disable this feature.

## Syntax Description

| | |
|---|---|
| <*number*> | Defines the DID number assigned to the operator group. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example assigns DID **44** to the operator group **1234**:

(config)**#voice operator-group 1234**
(config-operator-group)**#did 44**

# email <*address*>

Use the **email** command to enter the email address for this user's group. Use the **no** form of this command to disable this feature.

## Syntax Description

| | |
|---|---|
| <*address*> | Specifies an email dress for this group. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example creates an email contact for this group:

(config)**#voice ring-group 1234**
(config-1234)**#email admin@helpdesk.com**

# email-secondary *<address>*

Use the **email-secondary** command to enter a secondary email address for this user's group. Use the **no** form of this command to disable this feature.

## Syntax Description

| | |
|---|---|
| *<address>* | Specifies a contact email address for this group. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example creates a secondary email contact for this group:

(config)**#voice ring-group 1234**
(config-1234)**#email-secondary lead@helpdesk.com**

# login-member <*number*>

Use the **login-member** command to log an existing member of the operator group into the system. You must first use the member command to create a new group member. Use the **no** form of this command to disable this feature. Refer to *member* <*number*> on page 1804 for more information.

## Syntax Description

| | |
|---|---|
| <*number*> | Specifies the extension number of the user who is logging in. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Functional Notes

This command allows a user to log in and out of a operator group, letting the system know when a user is available to accept calls.

## Usage Examples

The following example logs in the user at extension **4422**:

(config)**#voice operator-group 1234**
(config-operator-group)**#login-member 4422**

# max-inbound <*value*>

Use the **max-inbound** command to define the maximum number of calls that can be inbound at the same time. Use the **no** form of this command to return to the default setting.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the maximum number of calls that can be inbound at the same time. Range is 1 to 10 calls. |

## Default Values

By default, the maximum number of inbound calls is set to 1.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example sets the maximum number of inbound calls on operator group **1234** to **3**:

(config)**#voice operator-group 1234**
(config-operator-group)**#max-inbound 3**

# member *<number>*

Use the **member** command to create a new member of the operator group. Use the **no** form of this command to remove a user's extension from a operator group.

## Syntax Description

| | |
|---|---|
| *<number>* | Specifies the extension number of the user you want to add as a operator group member. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Functional Note

A user can log in and out of the operator group using the **login-member** and **no login-member** commands. Refer to *email <address>* on page 1800 for more information.

## Usage Examples

The following example adds the user at extension **4422** to the operator group **1234**:

(config)**#voice operator-group 1234**
(config-operator-group)**#member 4422**

# num-rings <*value*>

Use the **num-rings** command to define the maximum number of rings allowed at each extension in the operator group before the call is forwarded as specified by the operator group's coverage list. Use the **no** form of this command to return to the default setting.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the maximum number of rings for the operator group. Valid range is 2 to 10 rings. |

## Default Values

By default, the maximum number of rings allowed at each extension is 2.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example sets the maximum number of rings for the operator group **1234** to **6**:

(config)**#voice operator-group 1234**
(config-operator-group)**#num-rings 6**

# prefix

Use the **prefix** command to turn on the caller ID prefix for this operator group, causing **GRP:** to display in front of the caller ID information. Use the **no** form of this command to turn the off prefix.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example turns on the caller ID prefix for operator group **1234**:

(config)**#voice operator-group 1234**
(config-operator-group)**#prefix**

# show voice mail

Use the **show voice mail** command to display information about the voice mail store for the users. Variations of this command include:

**show voice mail**
**show voice mail notify-schedule**

## Syntax Description

| | |
|---|---|
| **notify-schedule** | Optional. Displays the voice mail notification schedule day, time, and email contacts. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |
| Release 13.1 | Command was updated. |

## Usage Examples

The following example displays voice mail information for ring group **1234**:

(config)**#voice ring-group 1234**
(config-1234)**#show voice mail**

Voicemail information for account: 2002

VM Class of Service: normal_voicemail
Internal Greeting (min): 00:00 Total Voicemail Usage (min): 00:00
External Greeting (min): 00:00 Total Voicemail Free (min): 10:00
Temporary Greeting (min): 00:00 Recorded Name (min): 00:00

This user has No Voicemail Messages
NV7100(config-1212)#

# sip-identity

Use the **sip-identity** command to configure the Session Initiation Protocol (SIP) registration options for the user. Use the **no** form of the command to disable the settings. Variations of this command include the following:

**sip-identity** <*station*> <*Txx*>
**sip-identity** <*station*> <*Txx*> **register**
**sip-identity** <*station*> <*Txx*> **register auth-name** <*username*> **password** <*password*>

## Syntax Description

| | |
|---|---|
| <*station*> <*Txx*> | Specifies the station to be used for SIP trunk (e.g., station extension) and the SIP trunk (Txx, e.g. T01) through which to register the server. |
| **register** | Registers the user to the server. |
| **register auth-name** <*username*> | Optional. Sets the username that will be required as authentication for registration to the SIP server. |
| **password** <*password*> | Optional. Sets the password that will be required as authentication for registration to the SIP server. |

## Default Values

No default values necessary for this command.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |
| Release 13.1 | Command functionality was introduced to this section. |

## Usage Examples

The following example specifies trunk **T02** and extension **4400** for SIP **identity**:

(config)**#voice ring-group 1234**
(config-4444)**#sip identity 4400 T02**

# type

Use the **type** command to configure the group type for the operator group. Variations of this command include:

**type all**
**type executive**
**type linear**
**type ucd**

## Syntax Description

| | |
|---|---|
| **all** | Configures the group as an all-inclusive operator group. When a operator group call comes in, all phones ring simultaneously. |
| **executive** | Configures an executive operator group. Refer to *email <address>* on page 1815 for more information. |
| **linear** | Configures the group as a linear hunt operator group. Member phones ring one at a time until the call is picked up. When the next call comes in, the call cycle begins again by ringing the first operator group member. Refer to *member <number>* on page 1804 for more information. |
| **ucd** | Configures the group as a Uniform Call Distribution (UCD) operator group. Member phones ring one at a time until the call is picked up. When the next call comes in, the system remembers which member extension it last dialed and then continues the call cycle by ringing the next member in the operator group. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example configures the operator group **1234** to ring all phones in the operator group each time a call comes in:

(config)**#voice operator-group 1234**
(config-operator-group)**#type all**

# voicemail

Use the **voicemail** command to configure the voice mail options for the users. Use the **no** form of the command to disable the settings. Variations of this command include the following:

**voicemail cos** <*name*>
**voicemail notify email text-only**
**voicemail notify schedule** <*day*> <*time*> **[am | pm]**
**voicemail oper-assist**
**voicemail password**

## Syntax Description

| | |
|---|---|
| **cos** <*name*> | Configures the voicemail class of service (CoS) type by entering the name of the selected CoS. |
| **notify email text-only** | Specifies the email address to alert when a new voice mail is received. |
| **notify schedule** | Configures the daily schedules for voice mail notification. Different schedules can be programmed for different days of the week. |
| <*day*> | Specifies the day of the week for the voice mail notification. Choose from Monday through Sunday. |
| <*time*> **[am \| pm]** | Specifies the time for the voice mail notification. Time should be expressed in the format hours:minutes (HH:MM) a.m. or p.m. |
| **oper-assist** <*number*> | Directs all operator calls to the specified phone number |
| **password** <*password*> | Creates the password/pin that will be required to access voicemail. |

## Default Values

No default values necessary for this command.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example configures the voice mail notification schedule for this group on Mondays at 6:30 p.m. Thursdays at 6:30 p.m. :

(config)**#voice ring-group 1234**
(config-4444)**#voicemail notify schedule monday 06:30 pm**

# VOICE RING GROUP COMMAND SET

To activate the Voice Ring Group Interface Configuration mode, enter the **voice ring-group** command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#voice ring-group xxxx*
(config-xxxx)#
```

*where xxxx = the ring group's four-digit extension.

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:



All other commands for this command set are described in this section in alphabetical order.



# assistant-extension <*number*>

Use the **assistant-extension** command to tie an assistant's extension to an executive's extension.

*This command only applies to a ring groups of **type executive**. Refer to type on page 1826 for more information*

## Syntax Description

| | |
|---|---|
| <*number*> | Specifies the number of the assistant's extension. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Functional Notes

This command is used in conjunction with the **executive-extension** command (refer to *email <address>* on page 1815). When the executive's extension is dialed, both the assistant's and the executive's phones will ring. If neither phone is answered (or both are busy or set to do-not-disturb), the call is forwarded through the executive's call coverage list.

## Usage Examples

The following example creates the executive ring group **1234** and causes both the executive (extension **1234**) and the assistant (extension **4444**) phones to ring when the executive extension is dialed:

(config)**#voice ring-group 1234**
(config-1234)**#type executive**
(config-1234)**#executive-extension 1234**
(config-1234)**#assistant-extension 4444**

# coverage

Use the **coverage** command to configure call coverage parameters for the user. The setting of this command tells the system how to handle calls after a certain number of rings go unanswered. Use the **no** form of this command to disable coverage. Variations of this command include:

**coverage aa**
**coverage aa** <*number*>
**coverage external** <*number*>
**coverage external** <*number*> **num-ring** <*rings*>
**coverage internal** <*number*>
**coverage internal** <*number*> **num-ring** <*rings*>
**coverage vm**

## Syntax Description

| | |
|---|---|
| **aa** <*number*> | Forwards the phone to the auto attendant. It is optional to enter a specific extension programmed to forward. |
| **external** <*number*> | Forwards the phone to an external number. |
| **internal** <*number*> | Forwards the phone to an internal number. |
| **num-ring** <*rings*> | Optional. Select the number of rings for the external number before performing the next action. Valid range is 1 to 9 rings. |
| **vm** | Forwards the phone to voicemail. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |
| Release 11.1 | Command was updated to include the number of rings option. |
| Release 12.1 | Command was updated to include the auto attendant option. |
| Release 13.1 | Command was updated to include the internal and voicemail options. |

## Usage Examples

The following example specifies that the user's phone be forwarded to the internal extension **8500** after **3** rings.

(config)**#voice ring-group 1234**
(config-1234)**#coverage internal 8500 num-ring 3**

# **did** <*number*>

Use the **did** command to configure Direct Inward Dialing (DID) for this group. DID is used if a service provider is providing digits to the unit on inbound calls or if the unit needs to provide DID information to a piece of CPE equipment. Use the **no** form of this command to disable this feature.

## Syntax Description

| | |
|---|---|
| <*number*> | Defines the DID number assigned to the ring group. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example assigns DID **44** to the ring group **1234**:

(config)**#voice ring-group 1234**
(config-1234)**#did 44**

# email <*address*>

Use the **email** command to enter the email address for this user's group. Use the **no** form of this command to disable this feature.

## Syntax Description

| | |
|---|---|
| <*address*> | Specifies an email dress for this group. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example creates an email contact for this group:

(config)**#voice ring-group 1234**
(config-1234)**#email admin@helpdesk.com**

# email-secondary *<address>*

Use the **email-secondary** command to enter a secondary email address for this user's group. Use the **no** form of this command to disable this feature.

## Syntax Description

| | |
|---|---|
| *<address>* | Specifies a contact email address for this group. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example creates a secondary email contact for this group:

(config)**#voice ring-group 1234**
(config-1234)**#email-secondary lead@helpdesk.com**

# executive-extension *<number>*

Use the **executive-extension** command to tie an executive's extension to an assistant's extension.

*This command only applies to a ring groups of **type executive**. Refer to type on page 1826 for more information*

## Syntax Description

| | |
|---|---|
| *<number>* | Specifies the number of the executive's extension. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Functional Notes

This command is used in conjunction with the **assistant-extension** command (refer to *assistant-extension <number>* on page 1812). When the executive's extension is dialed, both the assistant's and the executive's phones will ring. If neither phone is answered (or both are busy or set to do-not-disturb), the call is forwarded through the executive's call coverage list.

## Usage Examples

The following example creates the executive ring group **1234** and causes both the executive (extension **1234**) and the assistant (extension **4444**) phones to ring when the executive extension is dialed:

(config)**#voice ring-group 1234**
(config-1234)**#type executive**
(config-1234)**#executive-extension 1234**
(config-1234)**#assistant-extension 4444**

# extension *<number>*

Use the **extension** command to change the extension for this ring group.

## Syntax Description

| | |
|---|---|
| *<number>* | Specifies a number for a new extension for the ring group. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example changes the extension of the ring group **1234** to **4321**:

(config)**#voice ring-group 1234**
(config-1234)**#extension 4321**
(config-4321)#

# login-member *<number>*

Use the **login-member** command to log an existing member of the ring group into the system. You must first use the member command to create a new group member. Use the **no** form of this command to disable this feature. Refer to *member <number>* on page 1821 for more information.

## Syntax Description

| | |
|---|---|
| *<number>* | Specifies the extension number of the user who is logging in. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Functional Notes

This command allows a user to log in and out of a ring group, letting the system know when a user is available to accept calls.

## Usage Examples

The following example logs in the user at extension **4422**:

(config)**#voice ring-group 1234**
(config-1234)**#login-member 4422**

# max-inbound <*value*>

Use the **max-inbound** command to define the maximum number of calls that can be inbound at the same time. Use the **no** form of this command to return to the default setting.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the maximum number of calls that can be inbound at the same time. Range is 1 to 10 calls. |

## Default Values

By default, the maximum number of inbound calls is set to 1.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example sets the maximum number of inbound calls on ring group **1234** to **3**:

(config)**#voice ring-group 1234**
(config-1234)**#max-inbound 3**

# member *<number>*

Use the **member** command to create a new member of the ring group. Use the **no** form of this command to remove a user's extension from a ring group.

## Syntax Description

| | |
|---|---|
| *<number>* | Specifies the extension number of the user you want to add as a ring group member. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Functional Note

A user can log in and out of the ring group using the **login-member** and **no login-member** commands. Refer to *login-member <number>* on page 1819 for more information.

## Usage Examples

The following example adds the user at extension **4422** to the ring group **1234**:

(config)**#voice ring-group 1234**
(config-1234)**#member 4422**

# num-rings <*value*>

Use the **num-rings** command to define the maximum number of rings allowed at each extension in the ring group before the call is forwarded as specified by the ring group's coverage list. Use the **no** form of this command to return to the default setting.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the maximum number of rings for the ring group. Valid range is 2 to 10 rings. |

## Default Values

By default, the maximum number of rings allowed at each extension is 2.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example sets the maximum number of rings for the ring group **1234** to **6**:

(config)**#voice ring-group 1234**
(config-1234)**#num-rings 6**

# prefix

Use the **prefix** command to turn on the caller ID prefix for this ring group, causing **GRP:** to display in front of the caller ID information. Use the **no** form of this command to turn the prefix off.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example turns on the caller ID prefix for ring group **1234**:

```
(config)#voice ring-group 1234
(config-1234)#prefix
```

# show voice mail

Use the **show voice mail** command to display information about the voice mail store for the users. Variations of this command include:

**show voice mail**
**show voice mail notify-schedule**

## Syntax Description

| | |
|---|---|
| **notify-schedule** | Optional. Displays the voice mail notification schedule day, time, and email contacts. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |
| Release 13.1 | Command was updated. |

## Usage Examples

The following example displays voice mail information for ring group **1234**:

(config)**#voice ring-group 1234**
(config-1234)**#show voice mail**

Voicemail information for account: 2002

VM Class of Service: normal_voicemail
Internal Greeting (min): 00:00 Total Voicemail Usage (min): 00:00
External Greeting (min): 00:00 Total Voicemail Free (min): 10:00
Temporary Greeting (min): 00:00 Recorded Name (min): 00:00

This user has No Voicemail Messages
NV7100(config-1212)#

# sip-identity

Use the **sip-identity** command to configure the Session Initiation Protocol (SIP) registration options for the user. Use the **no** form of the command to disable the settings. Variations of this command include the following:

**sip-identity** <*station*> <*Txx*>
**sip-identity** <*station*> <*Txx*> **register**
**sip-identity** <*station*> <*Txx*> **register auth-name** <*username*> **password** <*password*>

## Syntax Description

| | |
|---|---|
| <*station*> <*Txx*> | Specifies the station to be used for SIP trunk (e.g., station extension) and the SIP trunk (Txx, e.g. T01) through which to register the server. |
| **register** | Registers the user to the server. |
| **register auth-name** <*username*> | Optional. Sets the username that will be required as authentication for registration to the SIP server. |
| **password** <*password*> | Optional. Sets the password that will be required as authentication for registration to the SIP server. |

## Default Values

No default values necessary for this command.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example specifies trunk **T02** and extension **4400** for SIP **identity**:

(config)**#voice ring-group 1234**
(config-4444)**#sip identity 4400 T02**

# type

Use the **type** command to configure the group type for the ring group. Variations of this command include:

**type all**
**type executive**
**type linear**
**type ucd**

## Syntax Description

| | |
|---|---|
| **all** | Configures the group as an all-inclusive ring group. When a ring group call comes in, all phones ring simultaneously. |
| **executive** | Configures an executive ring group. Refer to *email <address>* on page 1815 for more information. |
| **linear** | Configures the group as a linear hunt ring group. Member phones ring one at a time until the call is picked up. When the next call comes in, the call cycle begins again by ringing the first ring group member. Refer to *member <number>* on page 1821 for more information. |
| **ucd** | Configures the group as a Uniform Call Distribution (UCD) ring group. Member phones ring one at a time until the call is picked up. When the next call comes in, the system remembers which member extension it last dialed and then continues the call cycle by ringing the next member in the ring group. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example configures the ring group **1234** to ring all phones in the ring group each time a call comes in:

(config)**#voice ring-group 1234**
(config-1234)**#type all**

# voicemail

Use the **voicemail** command to configure the voice mail options for the users. Use the **no** form of the command to disable the settings. Variations of this command include the following:

**voicemail cos** <*name*>
**voicemail notify email text-only**
**voicemail notify schedule** <*day*> <*time*> **[am | pm]**
**voicemail oper-assist**
**voicemail password**

## Syntax Description

| | |
|---|---|
| **cos** <*name*> | Configures the voicemail class of service (CoS) type by entering the name of the selected CoS. |
| **notify email text-only** | Specifies the email address to alert when a new voice mail is received. |
| **notify schedule** | Configures the daily schedules for voice mail notification. Different schedules can be programmed for different days of the week. |
| <*day*> | Specifies the day of the week for the voice mail notification. Choose from Monday through Sunday. |
| <*time*> **[am \| pm]** | Specifies the time for the voice mail notification. Time should be expressed in the format hours:minutes (HH:MM) a.m. or p.m. |
| **oper-assist** <*number*> | Directs all operator calls to the specified phone number |
| **password** <*password*> | Creates the password/pin that will be required to access voicemail. |

## Default Values

No default values necessary for this command.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example configures the voice mail notification schedule for this group on Mondays at 6:30 p.m. Thursdays at 6:30 p.m. :

(config)**#voice ring-group 1234**
(config-4444)**#voicemail notify schedule monday 06:30 pm**

# VOICE TRUNK ANALOG COMMAND SET

To activate the Voice Trunk Analog Dial Pulse Terminate (DPT) Configuration mode, enter the following command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#voice trunk t01 type analog supervision dpt
(config-t01)#
```

To activate the Voice Trunk Analog Loop Start (LS) Configuration mode, enter the following command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#voice trunk t01 type analog supervision loop-start
(config-t01)#
```

To activate the Voice Trunk Analog Ground Start (GS) Configuration mode, enter the following command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#voice trunk t01 type analog supervision ground-start
(config-t01)#
```


*Not all Trunk Analog commands apply to all analog trunk types. Use the ? command to display a list of valid commands.*

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:

*cross-connect* *on page 29*
*description <text>* *on page 32*
*do* *on page 33*
*exit* *on page 35*

All other commands for this command set are described in this section in alphabetical order.

*blind-dial* *on page 1830*
*busy all* *on page 1831*
*busy fxo <slot/port>* *on page 1832*
*busy range fxo <range>* *on page 1833*

# blind-dial

Use the **blind-dial** command to allow calls to be placed without the presence of dial-tone. Use the **no** form of this command to disable blind dialing.

## Syntax Description

No subcommands.

## Default Values

By default, **blind-dial** is disabled.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Usage Examples

The following example enables blind dialing:

```
(config)#voice trunk t01 type analog supervision loop-start
(config-t01)#blind-dial
```

# busy all

Use the **busy all** command to set all DS0s to busy so that no calls are allowed inbound or outbound. If any calls are active at the time this command is issued, the calls will stay active until either party terminates the call. Once terminated, the DS0s are busied out. Use the **no** form of this command to return to default settings. Variations of this command include:

**busy all**
**busy all now**

## Syntax Description

| | |
|---|---|
| **now** | Optional. Immediately terminates calls that are active at the time the command is issued (for example, in the middle of a conversation). |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |
| Release 11.1 | Command expanded to include the analog voice trunk. |

## Usage Examples

The following example sets all DS0s on trunk T01 to busy and terminates calls that are active at the time the command is issued:

(config)**#voice trunk t01 type analog supervision loop-start**
(config-t01)**#busy all now**

# busy fxo <*slot/port*>

Use the **busy fxo** command to set a DS0 to busy so that no calls are allowed inbound or outbound. If a call is active at the time this command is issued, the call will stay active until either party terminates the call. Once terminated, the DS0 is set to busy. Use the **no** form of this command to disable this feature. Variations of this command include:

**busy fxo** <*slot/port*>
**busy fxo** <*slot/port*> **now**

## Syntax Description

| | |
|---|---|
| <*slot/port*> | Specifies the slot/port for the FXO. |
| **now** | Optional. Immediately terminates active call at the time the command is issued (for example, in the middle of a conversation). |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |
| Release 11.1 | Command expanded to include the analog voice trunk. |

## Usage Examples

The following example sets FXO 0/1 to busy and terminates an active call when the command is issued:

(config)**#voice trunk t01 type analog supervision loop-start**
(config-t01)**#busy fxo 0/1 now**

# busy range fxo <*range*>

Use the **busy range fxo** command to set a particular set of DS0s to busy so that no calls are allowed inbound or outbound. If any calls are active at the time this command is issued, the calls will stay active until either party terminates the call. Once terminated, the DS0s are set to busy. Use the **no** form of this command to return to the default setting. Variations of this command include:

**busy range fxo** <*range*>
**busy range fxo** <*range*> **now**

## Syntax Description

| | |
|---|---|
| <*range*> | Specifies a range of ports in the format <*slot/begin port range-end port range*>. For example, **0/1-4**. |
| **now** | Optional. Terminates calls that are active at the time the command is issued (for example, in the middle of a conversation). |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example sets DS0s to busy and terminates assigned to port range FXO 0/1 through FXO 0/4 when the command is issued:

(config)**#voice trunk t01 type analog supervision loop-start**
(config-t01)**#busy range fxo 0/1-4 now**

# caller-id

Use the **caller-id** number command to interpret and pass caller identification (ID) on this trunk. This information usually displays the name, number, time and date of the calling party. Use the **no** form of this command to cancel the setting.

## Syntax Description

No subcommands.

## Default Values

By default, caller ID is disabled.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example enables caller ID:

(config)**#voice trunk t01 type analog supervision loop-start**
(config-t01)**#caller-id**

# caller-id-override number <*number*>

Use the **caller-id-override number** command to replace the calling party information for this trunk with a specific number. This command is used to conceal user's name and number or to display a different name and number for internal or external caller ID. Use the **no** form of this command to cancel the setting.Variations of this command include:

**caller-id-override number** <*number*>
**caller-id-override number** <*number*> <*trunk id*>

## Syntax Description

| | |
|---|---|
| <*number*> | Specifies the number to display on caller ID. |
| <*trunk id*> | Specifies the trunk id (Txx) for outbound calls. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example sets the caller ID override number at the time the command is issued:

(config)**#voice trunk t01 type analog supervision loop-start**
(config-t01)**#caller-id-override number 555-8000**

# codec-group *<name>*

Use the **codec-group** command to configure the CODEC groups to use for the account. You will need to define CODEC lists to specify CODEC order (refer to *voice codec-list <name>* on page 700 for information) prior to assigning the CODEC group. You will then use the *Voice CODEC List Configuration Command Set* on page 1747 to define the specific CODEC you want each group to use for negotiation. Use the **no** form of this command to remove assigned CODEC groups.

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies the name of the CODEC group you want to use for this trunk. |

## Default Values

By default, no CODEC groups are assigned to this trunk.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example associates the CODEC group named **user** with this trunk:

(config)**#voice trunk t01 type analog supervision loop-start**
(config-t01)**#codec-group user**

# connect fxo <*slot/port*>

Use the **connect fxo** command to specify the physical interface this trunk will use for voice calls. Use the **no** form of this command to return to the default setting.

## Syntax Description

| | |
|---|---|
| <*slot/port*> | Specifies the slot/port for the FXO trunk. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |
| Release 11.1 | Command expanded to include the analog voice trunk. |

## Usage Examples

The following example specifies this trunk will to use port FXO 0/1:

(config)**#voice trunk t01 type analog supervision loop-start**
(config-t01)**#connect fxo 0/1**

# connect range fxo *<range>*

Use the **connect range fxo** command to specify the range of physical interfaces for this trunk group usage. Use the **no** form of this command to return to the default setting.

## Syntax Description

| | |
|---|---|
| *<range* | Specifies a range of ports in the format *slot/begin port range-end port range*. For example, **0/1-4**. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example specifies that this analog loop start trunk will use the contiguous port range FXO 0/1 through FXO 0/4:

(config)**#voice trunk t01 type analog supervision loop-start**
(config-t01)**#connect range fxo 0/1-4**

# did digits-transferred <*value*>

Use the **did digits-transferred** command to define how many of the received digits should be sent to the internal switchboard from an incoming call on a User Role Trunk. The number of digits transferred are the least digits received. Direct Inward Dialing (DID) should be used if a Telco provider is providing digits to the unit on inbound calls or if the unit needs to provide DID information to a piece of CPE equipment. Use the **no** form of this command to disable this feature.Variations of this command include:

**did digits-transferred** <*value*>
**did digits-transferred** <*value*> **prefix** <*number*>

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the number of digits to be transferred. Range is 1 to 16 digits. |
| **prefix** <*number*> | Optional. Specifies a sequence of digits to be prepended to the digits that will be transmitted. For example, if seven digits will be transferred via DID, then prefix the seven digits with 256. Thus 555-8000 would be prefixed with 256, transmitting out the string of digits 256-555-8000. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Functional Notes

User Role Example:

555-1000 is an incoming call on the trunk. With **did** set to 4, the number 1000 should be sent to the switchboard. On a network role trunk, the **did** command allows you to define how many of the digits from the Accept criteria should be sent externally from a call that was routed by the switchboard. The number of digits transferred are the least significant digits received.

Network Role Example:

555-1000 is accepted on the UT interface. With DID Digits Transferred set to 4, the number of 1000 will be sent to the device connected to the UT interface. This command cannot be specified if and when **trunk-number** is being used. Conversely, if **did** is used, **trunk-number** will not be allowed.

## Usage Examples

The following example transfers the digits 555-8000 and adds the prefix 256:

(config)**#voice trunk t01 type analog supervision loop-start**
(config-t01)**#did digits-transferred 5558000 prefix 256**

# echo-cancellation

Use the **echo-cancellation** command to improve voice quality for packetized-based voice calls such as voice over IP (VoIP) or Media Gateway Control Protocol (MGCP). Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, **echo-cancellation** is enabled.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example activates **echo-cancellation**:

(config)**#voice trunk t01 type analog supervision loop-start**
(config-t01)**#echo-cancellation**

# match <*number*> substitute <*number*>

Use the **match substitute** command to substitute a different number for the number originally dialed by a user of the system. If no match occurs (or no match statements have been entered) the original dialed number will be propagated without being modified. Use the **no** form of this command to delete match substitutes.

*You may enter multiple match commands on each trunk. The first valid match that is found for outbound numbers will be used in configurations where more than one match statement might be valid for a given dialed number. Therefore, order of input is important.*

## Syntax Description

| | |
|---|---|
| **match** <*number*> | Specifies the dialed number that you are trying to match. |
| **substitute** <*number*> | Specifies the number that will be sent in place of the number that was matched. |

## Default Values

By default, no substitutions are defined.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example attempts to match the dialed number 555-8000 and specifies that number 555-8500 will be sent it no match occurs:

(config)**#voice trunk t01 type analog supervision loop-start**
(config-t01)**#match 5558000 substitute 5558500**

# modem-passthrough

Use the **modem-passthrough** command to switch to pass-through mode on fax or modem tone detection. This commands allows modem and fax calls to maintain a connection without altering the signals with the voice improvement settings such as echo cancellation and voice activity detection (vad). Use the **no** form of this command to disable this feature. Variations of this command include:

**modem-passthrough**
**modem-passthrough detection-time** <*value*>

## Syntax Description

| | |
|---|---|
| **detection-time** <*value*> | Optional. Specifies the fax and/or modem detection time length value in seconds. Range is 0 to 8 seconds. |

## Default Values

By default, **modem-passthrough** is enabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |
| Release 12.1 | Command was expanded. |

## Usage Examples

The following example disables **modem-passthrough**:

(config)**#voice trunk t01 type analog supervision loop-start**
(config-t01)**#no modem-passthrough**

# plc

Use the **plc** command to enable packet loss concealment (PLC). PLC is used to prevent choppy connections by concealing a packet loss by replacing the lost packet with another voice packet in the data stream. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, PLC is enabled on this interface.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example disables PLC on trunk T01:

(config)**#voice trunk t01 type analog supervision loop-start**
(config-t01)**#no plc**

# reject-external

Use the **reject-external** command to prevent inbound calls on the trunk from being routed back out of the same trunk. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, **reject-external** is enabled on this interface.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Functional Notes

In general, trunks are assigned to the user role, which means they terminate lines from a Telco provider. If this is the case, **reject-external** should be enabled so that inbound calls on the trunk cannot be routed back out of the same trunk. If the configuration is poor, inbound long distance calls could be routed back out the same trunk, causing the owner of the unit to be charged for long distance calls without his knowledge. For network-role trunks and SIP-based trunks, this command should be disabled to allow calls to be properly routed in the unit.

## Usage Examples

The following example disables **reject-external**:

(config)**#voice trunk t01 type analog supervision loop-start**
(config-t01)**#no reject-external**

# resource-selection

Use the **resource-selection** command to determine how the switchboard uses outbound call resources contained within a TDM-based trunk group. Use the **no** form of this command to disable this feature. Variations of this command include:

**resource-selection circular**
**resource-selection circular ascending**
**resource-selection circular descending**
**resource-selection linear**
**resource-selection linear ascending**
**resource-selection linear descending**

## Syntax Description

| | |
|---|---|
| **circular** | Performs call load balancing among available DS0s/B-channels in this trunk. Subsequent calls will be delivered to the next available DS0/B-channel in a round-robin fashion. |
| **linear** | Specifies that a call being delivered to this trunk will be accepted out the first available DS0/B-channel available at the time the call is received. |
| **ascending** | Optional. Distributes calls in an order from the lowest to the highest channel (DS0 1, 2, 3 through 24). |
| **descending** | Optional. Distributes calls in an order from the highest to the lowest channel (DS0 24, 23, 22 through 1). |

## Default Values

By default, resource selection is set to **linear**.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |
| Release 13.1 | Command was expanded to include the ascending and descending subcommands. |

## Usage Examples

The following example specifies circular resource selection:

(config)**#voice trunk t01 type t1-rbs supervision wink role network**
(config-T01)**#resource-selection circular**

# rtp delay-mode

Use the **rtp delay-mode** command to configure the Realtime Transport Protocol (RTP) jitter buffer packet delay mode settings. RTP is used to prevent static voice connections by enhancing the quality of the packet delivery. Use the **no** form of this command to return to the default settings. Variations of this command include:

**rtp delay-mode adaptive**
**rtp delay-mode fixed**

## Syntax Description

| | |
|---|---|
| **adaptive** | Configures the RTP jitter buffer packet delay to adjust during a call based on network conditions. |
| **fixed** | Configures the RTP jitter buffer packet delay to remain constant. |

## Default Values

By default, the RTP delay mode is set to **adaptive**. This allows for minimal latency by adjusting the average packet delay based on the conditions of the network

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example configures RTP delay mode as fixed:

(config)**#voice trunk t01 type analog supervision loop-start**
(config-t01)**#rtp delay-mode fixed**

# rtp dtmf-relay

Use the **rtp dtmf-relay** command to configure the method by which Realtime Transport Protocol (RTP) Dual Tone Multi-Frequency (DTMF) events are relayed. The dial digits can be sent inband or out-of-band of the voice stream. Use the **no** form of this command to return to the default value. Variations of this command include:

**rtp delay-mode adaptive**
**rtp delay-mode fixed**

## Syntax Description

| | |
|---|---|
| **inband** | Specifies that RTP DTMF events be relayed inband in the RTP stream. |
| **nte** <*value*> | Specifies that RTP DTMF events be relayed out-of-band using NTE. Enter an NTE value between 96 and 127. |

## Default Values

By default, the **rtp dtmf-relay** is set for NTE 101.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example configures RTP DTMF relay events for **inband**:

(config)**#voice trunk t01 type analog supervision loop-start**
(config-t01)**#rtp dtmf-relay inband**

# rtp frame-packetization <*value*>

Use the **rtp frame-packetization** command to configure the Realtime Transport Protocol (RTP) frame packetization time in milliseconds for individual trunks and users. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Configures the RTP frame packetization time value in milliseconds. Select from 10, 20, or 30 milliseconds. |

## Default Values

By default, the **rtp frame-packetization** time is set to 20 milliseconds on all trunks and users.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example sets the frame packetization time for trunk T01 to 10 milliseconds:

(config)**#voice trunk t01 type analog supervision loop-start**
(config-t01)**#rtp frame-packetization 10**

# rtp packet-delay

Use the **rtp packet-delay** command to configure the maximum Realtime Transport Protocol (RTP) packet delays. This command is used to set the allowable limits of latency on the network. Use the **no** form of this command to return to default values. Use the **no** form of this command to return to default values. Variations of this command include:

**rtp packet-delay fax** <*value*>
**rtp packet-delay maximum** <*value*>
**rtp packet-delay nominal** <*value*>

## Syntax Description

| | |
|---|---|
| **fax** <*value*> | Sets the fax delay time value. Range is 0 to 500 milliseconds. |
| **maximum** <*value*> | Sets the maximum delay time value in increments of 10 milliseconds. Range is 40 to 320 milliseconds. |
| **nominal** <*value*> | Sets the nominal delay time value in increments of 10 milliseconds. Range is 10 to 240 milliseconds. |

## Default Values

By default, the RTP packet delay for fax is 300, maximum is 100, and nominal is 50.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example configures the RTP fax delay time on trunk T01 to **200** milliseconds:

(config)**#voice trunk t01 type analog supervision loop-start**
(config-t01)**#rtp packet-delay fax 200**

# rtp qos dscp <*value*>

Use the **rtp qos dscp** command to configure the maximum Realtime Transport Protocol (RTP) quality of service (QoS) parameters for differentiated services code point (DSCP). Use the **no** form of this command to return to default values.

## Syntax Description

| | |
|---|---|
| <*value*> | Configures the RTP QoS parameter for DSCP. Enter a value between 10 and 63. |

## Default Values

By setting the **rtp qos dscp** value on an individual trunk or user, you will override the global **rtp qos dscp** setting for RTP packets. QoS is set using a Differentiated Services Code Point, or DSCP value. Valid DSCP values are 10-63, and a higher DSCP value has a higher priority. The default DSCP value for RTP is 46. Remember that if you are using a public IP connection, such as the Internet, for voice over IP, end-to-end QoS may not be guaranteed. The default DSCP value for SIP is 26. To configure QoS for the RTP traffic that carries the voice conversation, use the command, **ip rtp qos dscp** followed by the desired DSCP value.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Functional Note

By setting the **rtp qos dscp** value on an individual trunk or user, you will override the global **rtp qos dscp** setting for RTP packets.

## Usage Examples

The following example configures the RTP QoS DSCP for trunk T02 to **46**:

(config)**#voice trunk t01 type analog supervision loop-start**
(config-t01)**#rtp qos dscp 46**

# trunk-number <*number*>

Use the **trunk number** command to define the call routing when DID is disabled. This feature directs incoming calls to the specified number when DID is not present. Use the **no** form of this command to disable this feature.

## Syntax Description

| | |
|---|---|
| <*number*> | Specify the number used for call routing when DID is disabled. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example defines call routing on trunk T01:

(config)**#voice trunk t01 type analog supervision loop-start**
(config)**#trunk-number 4000**

# vad

Use the **vad** command to enable voice activity detection (VAD). VAD blocks out noise categorized as silence during a voice connection. The silent voice packets are not transmitted, allowing bandwidth usage to be reduced. Although VAD saves bandwidth, the quality of the voice call may be compromised. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, VAD is enabled for all T1 RBS trunks and users.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example disables VAD on trunk T02:

(config)**#voice trunk t01 type analog supervision loop-start**
(config-t01)**#no vad**

# VOICE TRUNK GROUP COMMAND SET

To activate the Voice Trunk Group Configuration mode, enter the **voice grouped-trunk** command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#voice grouped-trunk TestGroupvoice grouped-trunk TestGroup
(config-TestGroup)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:

*cross-connect* *on page 29*
*description <text>* *on page 32*
*do* *on page 33*
*exit* *on page 35*

All other commands for this command set are described in this section in alphabetical order.

*accept <pattern>* *on page 1854*
*reject <pattern>* *on page 1856*
*resource-selection* *on page 1857*
*trunk <Txx>* *on page 1858*

# accept *<pattern>*

Use the **accept** command to specify numbers that users can dial on the trunk. This command controls the type of outbound calls users can place on the system. Use the **no** form of this command to remove a configured dial pattern and return to the default setting. Variations of this command include:

**accept** <*pattern*>
**accept** <*pattern*> **cost** <*value*>

## Syntax Description

| | |
|---|---|
| *<pattern>* | Specifies the patterns users can dial on the trunk. You can use wildcards to help define accepted numbers. Wildcards are N= 2 to 9, M = 1 to 8, X = 0 to 9, and [1, 2, 3]. |
| **cost** <*value*> | Specifies the cost value for the trunk. This option is used if a call is accepted by several trunks. The call will be routed to the trunk with the lowest cost value. |

## Default Values

By default, the cost value is zero.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Functional Notes

The available wildcards for this command are:

555-8123 = Any digit matches only itself.

X = Any single digit (0 to 9).

N = Any digit 2 to 9.

$ = Any number of digits of any value.

[1, 2, 3] = Any single digit in this list.

The special characters ( ), -, + are always ignored.

Examples: 1) 555-81XX matches 555-8100 to 555-8199.

2) 555-812[0,1,2] matches 555-8120 to 555-8122.

3) 1-800$ matches any 1-800 calls.

4) Nxx-xxxx matches 7 digit local.

5) 1-Nxx-Nxx-xxxx matches LD calls North America.

## Usage Examples

The following example allows users on the trunk **TestGroup** to dial any local number:

(config)**#voice grouped-trunk TestGroup**
(config-TestGroup)**#accept Nxxxxxx**

# reject *<pattern>*

Use the **reject** command to specify numbers users cannot dial on the trunk. This features allows administrators to restrict callers from unwanted outbound calls such as international calls and 900 numbers. Use the **no** form of this command to disable this feature.

## Syntax Description

| | |
|---|---|
| *<pattern>* | Specifies the patterns that users cannot dial on the trunk. You can use wildcards to help define rejected numbers. Wildcards are N = 2 to 9, M = 1 to 8, X = 0 to 9, and [1, 2, 3]. For example, you can enter **900$** to prevent users from dialing all 900 numbers. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Functional Notes

The available wildcards for this command are:

555-8123 = Any digit matches only itself.

X = Any single digit (0 to 9).

N = Any digit 2 to 9.

$ = Any number of digits of any value.

[1, 2, 3] = Any single digit in this list.

Punctuation characters ( ), -, + are always ignored.

Examples: 1) 555-81XX matches 555-8100 to 555-8199.

2) 555-812[0,1,2] matches 555-8120 to 555-8122.

3) 1-800$ matches any 1-800 calls.

4) Nxx-xxxx matches 7 digit local.

5) 1-Nxx-Nxx-xxxx matches LD calls North America.

## Usage Examples

The following example blocks calls to any 900 number on the trunk **TestGroup**:

(config)**#voice grouped-trunk TestGroup**
(config-TestGroup)**#reject 900$**

# resource-selection

Use the **resource-selection** command to determine how the switchboard uses outbound call resources contained within a TDM-based trunk group. Use the **no** form of this command to disable this feature. Variations of this command include:

**resource-selection circular**
**resource-selection circular ascending**
**resource-selection circular descending**
**resource-selection linear**
**resource-selection linear ascending**
**resource-selection linear descending**

## Syntax Description

| | |
|---|---|
| **circular** | Performs call load balancing among available DS0s/B-channels in this trunk. Subsequent calls will be delivered to the next available DS0/B-channel in a round-robin fashion. |
| **linear** | Specifies that a call being delivered to this trunk will be accepted out the first available DS0/B-channel available at the time the call is received. |
| **ascending** | Optional. Distributes calls in an order from the lowest to the highest channel (DS0 1, 2, 3 through 24). |
| **descending** | Optional. Distributes calls in an order from the highest to the lowest channel (DS0 24, 23, 22 through 1). |

## Default Values

By default, resource selection is set to **linear**.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |
| Release 13.1 | Command was expanded to include the ascending and descending subcommands. |

## Usage Examples

The following example specifies circular resource selection:

(config)**#voice trunk t01 type t1-rbs supervision wink role network**
(config-T01)**#resource-selection circular**

## trunk <*Txx*>

Use the **trunk** command to add an existing trunk to the trunk group so outbound calls may be placed out that particular trunk as well. Use the **no** form of this command to remove a configured trunk group.

### Syntax Description

| | |
|---|---|
| <*Txx*> | Specifies an ID number for the trunk. The trunk ID is in the format Txx where xx is the trunk ID number. Enter a trunk ID between 1 and 99. For example, trunk **T02**. |

### Default Values

No default value necessary for this command.

### Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

### Usage Examples

The following example adds trunk T02 to the trunk group:

(config)**#voice grouped-trunk TestGroup**
(config)**#trunk t02**

# VOICE TRUNK ISDN COMMAND SET

To activate the Voice Trunk ISDN Configuration mode, enter **voice trunk txx type isdn** at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#voice trunk txx type isdn
(config-t01)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:

*cross-connect on page 29*
*description <text> on page 32*
*exit on page 35*

All other commands for this command set are described in this section in alphabetical order.



# caller-id-override number <*number*>

Use the **caller-id-override number** command to replace the calling party information for this trunk with a specific number. This command is used to conceal the user's name and number or to display a different name and number for internal or external caller ID. Use the **no** form of this command to cancel the setting. Variations of this command include:

**caller-id-override number** <*number*>
**caller-id-override number** <*number*> <*trunk id*>

## Syntax Description

| | |
|---|---|
| <*number*> | Specifies the number to display on caller ID. |
| <*trunk id*> | Specifies the trunk id (Txx) for outbound calls. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |
| Release 11.1 | Command expanded to include the ISDN trunk. |
| Release 12.1 | Command was updated. |

## Usage Examples

The following example sets the caller-ID number to appear as 555-8000 to the receiving party at the time the command is issued:

(config)**#voice trunk t01 type isdn**
(config-t01)**#caller-id-override number 555-8000**

# codec-group *<name>*

Use the **codec-group** command to specify the CODEC group to use for this trunk. For information on defining CODEC groups, refer to *codec-group <name>* on page 1927. Use the **no** form of this command to return to the default setting.

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies the name of the CODEC group you want to use. |

## Default Values

By default, no CODEC groups are assigned to this interface.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |
| Release 11.1 | Command expanded to include the ISDN trunk. |

## Usage Examples

The following example associates the CODEC group **user** with this trunk:

(config)**#voice trunk t01 type isdn**
(config-T01)**#codec-group user**

# connect isdn-group <*value*>

Use the **connect isdn-group** command to associate a trunk with an ISDN group. The ISDN group number uniquely identifies an ISDN trunk group. Use the **no** form of this command to remove this association.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the ISDN group number. Range is 1 to 255. |

## Default Values

By default, no group is defined.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |
| Release 11.1 | Command expanded to include the ISDN trunk. |

## Usage Examples

The following example specifies that this trunk will use the ISDN group 1:

(config)**#voice trunk t01 type isdn**
(config-T01)**#connect isdn-group 1**

# echo-cancellation

Use the **echo-cancellation** command to improve voice quality for packetized-based voice calls such as Voice over IP (VoIP) or Media Gateway Control Protocol (MGCP). Enabling this command may significantly improve the voice quality in calls across the telephone network. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, echo cancellation is enabled.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |
| Release 11.1 | Command expanded to include the ISDN trunk. |

## Usage Examples

The following example activates **echo-cancellation**:

(config)**#voice trunk t01 type isdn**
(config-T01)**#echo-cancellation**

# match <*number*> substitute <*number*>

Use the **match substitute** command to substitute a different number for the number originally dialed by a user of the system. If no match occurs (or no match statements have been entered) the original dialed number will be propagated without being modified. Use the **no** form of this command to delete match substitutions.

*You may enter multiple match commands on each trunk. The first valid match that is found for outbound numbers will be used in configurations where more than one match statement might be valid for a given dialed number. Therefore, order of input is important.*

## Syntax Description

| | |
|---|---|
| **match** <*number*> | Specifies the dialed number that you are trying to match. |
| **substitute** <*number*> | Specifies the number that will be sent in place of the number that was matched. |

## Default Values

By default, no substitutions are defined.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |
| Release 11.1 | Command expanded to include the ISDN trunk. |

## Usage Examples

The following example attempts to match the dialed number 555-8000 and specifies that number 555-8500 will be sent it no match occurs:

(config)**#voice trunk t01 type isdn**
(config-T01)**#match 5558000 substitute 5558500**

# modem-passthrough

Use the **modem-passthrough** command to switch to pass-through mode on fax or modem tone detection. This commands allows modem and fax calls to maintain a connection without altering the signals with the voice improvement settings such as echo cancellation and voice activity detection (vad). Use the **no** form of this command to disable this feature. Variations of this command include:

**modem-passthrough**
**modem-passthrough detection-time** <*value*>

## Syntax Description

| | |
|---|---|
| **detection-time** <*value*> | Optional. Specifies the fax and/or modem detection time length value in seconds. Range is 0 to 8 seconds. |

## Default Values

By default, **modem-passthrough** is enabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |
| Release 12.1 | Command was expanded. |

## Usage Examples

The following example enables **modem-passthrough**:

(config)**#voice trunk t01 type isdn**
(config-T01)**#modem-passthrough**

# plc

Use the **plc** command to enable packet loss concealment (PLC). PLC is used to prevent choppy connections by concealing a packet loss by replacing the lost packet with another voice packet in the data stream. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, PLC is enabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example enables PLC on trunk T01:

```
(config)#voice trunk t01 type isdn
(config-t01)#plc
```

# reject-external

Use the **reject-external** command to blocked outbound (external) call attempts. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, this command is enabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example enables PLC on trunk T01:

(config)**#voice trunk t01 type isdn**
(config-t01)**#reject-external**

# resource-selection

Use the **resource-selection** command to determine how the switchboard uses outbound call resources contained within a TDM-based trunk group. Use the **no** form of this command to disable this feature. Variations of this command include:

**resource-selection circular**
**resource-selection circular ascending**
**resource-selection circular descending**
**resource-selection linear**
**resource-selection linear ascending**
**resource-selection linear descending**

## Syntax Description

| | |
|---|---|
| **circular** | Performs call load balancing among available DS0s/B-channels in this trunk. Subsequent calls will be delivered to the next available DS0/B-channel in a round-robin fashion. |
| **linear** | Specifies that a call being delivered to this trunk will be accepted out the first available DS0/B-channel available at the time the call is received. |
| **ascending** | Optional. Distributes calls in an order from the lowest to the highest channel (DS0 1, 2, 3 through 24). |
| **descending** | Optional. Distributes calls in an order from the highest to the lowest channel (DS0 24, 23, 22 through 1). |

## Default Values

By default, resource selection is set to **linear**.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |
| Release 13.1 | Command was expanded to include the ascending and descending subcommands. |

## Usage Examples

The following example specifies circular resource selection:

(config)**#voice trunk t01 type t1-rbs supervision wink role network**
(config-T01)**#resource-selection circular**

# rtp delay-mode

Use the **rtp delay-mode** command to configure the Realtime Transport Protocol (RTP) jitter buffer packet delay mode settings. RTP is used to prevent static voice connections by enhancing the quality of the packet delivery. Use the **no** form of this command to return to the default settings. Variations of this command include:

**rtp delay-mode adaptive**
**rtp delay-mode fixed**

## Syntax Description

| | |
|---|---|
| **adaptive** | Configures RTP jitter buffer packet delay to adjusts during a call based on network conditions. |
| **fixed** | Configures RTP jitter buffer packet delay remains constant. |

## Default Values

By default, this command is set to **adaptive**.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example configures RTP delay mode as adaptive:

(config)**#voice trunk t01 type isdn**
(config-T02)**#rtp delay-mode adaptive**

# rtp dtmf-relay

Use the **rtp dtmf-relay** command to configure the method by which Realtime Transport Protocol (RTP) Dual Tone Multi-Frequency (DTMF) events are relayed, either inband in the RTP stream or out-of-band using Named Telephone Events (NTE). RTP DTMF relay is used to prevent the tone (dialed digits) from being corrupted. Use the **no** form of this command to return to the default value. Variations of this command include:

**rtp dtmf-relay inband**
**rtp dtmf-relay nte** <*value*>

## Syntax Description

| | |
|---|---|
| **inband** | Configures RTP DTMF relay events for inband. |
| **nte** <*value*> | Configures RTP DTMF relay events for NTE. Enter a value between 96 and 127. |

## Default Values

By default, the **rtp dtmf-relay** is set for NTE 101.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |
| Release 11.1 | Command expanded to include the ISDN trunk. |

## Usage Examples

The following example configures RTP DTMF relay events for NTE with an event value of 101:

(config)**#voice trunk t01 type isdn**
(config-T02)**#rtp dtmf-relay nte 101**

# rtp frame-packetization <*value*>

Use the **rtp frame-packetization** command to configure the Realtime Transport Protocol (RTP) frame packetization time in milliseconds for individual trunks and users. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Configures the RTP frame packetization time in milliseconds. Select from 10, 20, or 30 milliseconds. |

## Default Values

By default, the **rtp frame-packetization** time is set to 20 milliseconds on all trunks and users.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |
| Release 11.1 | Command expanded to include the ISDN trunk. |

## Usage Examples

The following example sets the frame packetization time for trunk T01 to 20 milliseconds:

(config)**#voice trunk t01 type isdn**
(config-t01)**#rtp frame-packetization 20**

# rtp packet-delay

Use the **rtp packet-delay** command to configure the maximum Realtime Transport Protocol (RTP) packet delays. This command is used to set the allowable limits of latency on the network. Use the **no** form of this command to return to default values. Use the **no** form of this command to return to default values. Variations of this command include:

**rtp packet-delay fax** <*value*>
**rtp packet-delay maximum** <*value*>
**rtp packet-delay nominal** <*value*>

## Syntax Description

| | |
|---|---|
| **fax** <*value*> | Sets the fax delay time in milliseconds. Range is 0 to 500 milliseconds. |
| **maximum** <*value*> | Sets the maximum delay time in increments of 10 milliseconds. Range is 40 to 320 milliseconds. |
| **nominal** <*value*> | Sets the nominal delay time in increments of 10 milliseconds. Range is 10 to 240 milliseconds. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example configures the RTP fax delay time on trunk T01 to 10 milliseconds:

(config)**#voice trunk t01 type isdn**
(config-T01)**#rtp packet-delay fax 10**

# rtp qos dscp <*value*>

Use the **rtp qos dscp** command to configure the maximum Realtime Transport Protocol (RTP) quality of service (QoS) parameters for differentiated services code point (DSCP).

## Syntax Description

| | |
|---|---|
| <*value*> | Configures the RTP QoS parameter for differentiated services code point. Enter a value between 0 and 63. |

## Default Values

By default, no RTP QoS DSCP is configured for this interface.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |
| Release 11.1 | Command expanded to include the ISDN trunk. |

## Functional Note

By setting the **rtp qos dscp** value on an individual trunk or user, you will override the global **rtp qos dscp** setting for RTP packets. QoS is set using a Differentiated Services Code Point, or DSCP value. Valid DSCP values are 10-63, and a higher DSCP value has a higher priority. The default DSCP value for RTP is 46. Remember that if you are using a public IP connection, such as the Internet, for voice over IP, end-to-end QoS may not be guaranteed. The default DSCP value for SIP is 26. To configure QoS for the RTP traffic that carries the voice conversation, use the command, **ip rtp qos dscp** followed by the desired DSCP value.

## Usage Examples

By setting the **rtp qos dscp** value on an individual trunk or user, you will override the global **rtp qos dscp** setting for RTP packets. The following example sets the DSCP value for RTP packets that trunk T02 generates to **46**.

(config)**#voice trunk t01 type isdn**
(config-T02)**#rtp qos dscp 46**

# trunk-number <*number*>

Use the **trunk number** command to define the call routing when DID is disabled. This feature directs incoming calls to the specified number when DID is not present. Use the **no** form of this command to disable this feature.

## Syntax Description

| | |
|---|---|
| <*number*> | Specify the number used for call routing when DID is disabled. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example sets the call routing DID on trunk T01 to 4000:

(config)**#voice trunk t01 type isdn**
(config)**#trunk-number 4000**

# vad

Use the **vad** command to enable voice activity detection (VAD). VAD blocks out noise categorized as silence during a voice connection. The silent voice packets are not transmitted, allowing bandwidth usage to be reduced. Although VAD saves bandwidth, the quality of the voice call may be compromised. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, VAD is enabled for all T1 RBS trunks and users.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |
| Release 11.1 | Command expanded to include the ISDN trunk. |

## Usage Examples

The following example enables voice activation detection on trunk T01:

(config)**#voice trunk t01 type isdn**
(config-t01)**#vad**

# VOICE TRUNK SIP COMMAND SET

To activate the Voice Trunk Session Initiation Protocol (SIP) Interface Configuration mode, enter the **voice trunk type sip** command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#voice trunk T01 type sip
(config-T01)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:

*cross-connect on page 29*
*description <text> on page 32*
*do on page 33*
*exit on page 35*

All other commands for this command set are described in this section in alphabetical order.



# authentication username <*username*> password <*password*>

Use the **authentication** command to enable authentication security between the Session Initiation Protocol (SIP) server and the unit. Each port that registers with the SIP server will use the defined **username** and **password**. Use the **no** form of this command to return to default settings.

## Syntax Description

| | |
|---|---|
| **username** <*username*> | Specifies a string to be sent as the user name in authentication. |
| **password** <*password*> | Specifies a string to be sent as the password in authentication. |

## Default Values

By default, authentication is not enabled.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Functional Notes

If all users on the trunk use the same user name/password, enter the user name and password for authentication under the trunk. Otherwise, enter authentication information for each user individually in the Voice User command set which overrides the setting of this command. Refer to *Voice User Configuration Command Set* on page 1922 for more information.

## Usage Examples

The following example configures a user name of **iaduser** and password of **totalaccess** at the trunk level:

(config)**#voice trunk T01 type sip**
(config-T01)**#authentication username iaduser password totalaccess**

# caller-id-override number <*number*>

Use the **caller-id-override number** command to change the caller identification (ID) display number for this trunk with a number of your choice. This command is used to conceal the user's original number. Use the **no** form of this command to remove the caller ID override feature. Variations of this command include:

**caller-id-override number** <*number*>
**caller-id-override number** <*number*> <*trunk id*>

## Syntax Description

| | |
|---|---|
| <*number*> | Specifies the number to display on caller ID. |
| <*trunk id*> | Specifies the trunk id (Txx) for outbound calls. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example displays the number **4422** on the caller ID display for this trunk:

(config)**#voice trunk T01 type sip**
(config-T01)**#caller-id-override number 4422**

# codec-group *<name>*

Use the **codec-group** command to apply a previously created CODEC list to the Session Initiation Protocol (SIP) trunk. Use the **no** form of this command to return to default settings.

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies the name of the CODEC list you wish to apply to the trunk. |

## Default Values

By default, there is no CODEC group configured.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Functional Notes

Order is important when creating the CODEC list. The trunk attempts to use the first CODEC in the list to negotiate a call. If the first CODEC negotiation is unsuccessful, the trunk uses the second CODEC in the list. If this is unsuccessful, the call will fail. For information on defining CODEC groups, refer to *codec-group <name>* on page 1927.

## Usage Examples

The following example sets the CODEC group to **729only** (a CODEC list that has already been configured):

(config)**#voice trunk T01 type sip**
(config-T01)**#codec-group 729only**

# conferencing-uri *<value>*

Use the **conferencing-uri** command to configure a conference application server URI that controls and uniquely identifies a conference. Use the **no** form of this command to return to default settings.

## Syntax Description

| | |
|---|---|
| *<value>* | Specifies the extension or complete URI of the conference application server. |

## Default Values

By default, **conferencing-uri** is not configured.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example sets the **conferencing-uri** to **0606**:

(config)**#voice trunk T01 type sip**
(config-T01)**#conferencing-uri 0606**

# dial-string source

Use the **dial-string source** command to set the SIP dialing string field of your choice. Use the **no** form of this command to remove the specified setting. Variations of this command include the following:

**dial-string source request-uri**
**dial-string source to**

## Syntax Description

| | |
|---|---|
| **request-uri** | Specifies the Request URI user field as the dialing string source. |
| **to** | Specifies the To header field as the dialing string source. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example enables the To header user field as the **dial-string** for this trunk:

(config)**#voice trunk T01 type sip**
(config-T01)**#dial-string source to**

# domain *<name>*

Use the **domain** command to configure the assigned domain name for host messages. The domain is an unique identifier for the SIP users on the trunk. Use the **no** form of this command to disable this feature.

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies the domain name for the Session Initiation Protocol (SIP) trunk commands. |

## Default Values

By default, no domain is configured.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example sets the domain name as **home.com**:

(config)**#voice trunk T01 type sip**
(config-T01)**#domain home.com**

# match <*number*> substitute <*number*>

Use the **match substitute** command to substitute a different number for the number originally dialed by a user. If no match occurs (or no match statements have been entered) the original dialed number is propagated without being modified. Use the **no** form of this command to delete match substitutions.

*You can enter multiple match commands on each trunk. If multiple valid match statements exist for a given dialed number, the first rated match is used. Therefore, order of input is important.*

## Syntax Description

| | |
|---|---|
| **match** <*number*> | Specifies the dialed number you are trying to match. |
| **substitute** <*number*> | Specifies the number to be sent in place of the first argument matched. |

## Default Values

By default, no substitutions are defined.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example attempts to match the dialed number 555-8000 and specifies that number 555-8500 be sent it no match occurs:

(config)**#voice trunk t01 type sip**
(config-T01)**#match 5558000 substitute 5558500**

## max-number-calls <*value*>

Use the **max-number-calls** command to configure the maximum number of calls allowed on this trunk. This command is useful in controlling the call usage of the trunk. Use the **no** form of this command to return to default settings.

### Syntax Description

| | |
|---|---|
| <*value*> | Specifies the maximum number of calls allowed on this trunk. Range is 1 to 64 calls. |

### Default Values

By default, no maximum number of calls is specified.

### Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

### Usage Examples

The following example sets the maximum number of calls allowed to **25**:

(config)**#voice trunk T01 type sip**
(config-T01)**#max-num-calls 25**

# outbound-proxy primary <*value*>

Use the **outbound-proxy primary** command to define the primary name/address of the Session Initiation Protocol (SIP) proxy server to which the trunk will send all SIP messages. Use the **no** form of this command to return to default settings. Variations of this command include:

**outbound-proxy primary** <*value*>
**outbound-proxy primary** <*value*> **udp** <*number*>

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the fully qualified domain name (FQDN) or IP address of the outbound proxy server. |
| **udp** <*number*> | Optional. Sets the UDP port of the outbound proxy server. Port number range is 0 to 65,535. |

## Default Values

By default, the IP address is set to 0.0.0.0 and the UDP port is set to 5060.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Functional Notes

The configured value must resolve to a valid IP address.

## Usage Examples

The following example sets the outbound proxy server to **sip-proxy.adtran.com** with a UDP port of **2222**:

(config)**#voice trunk T01 type sip**
(config-T01)**#outbound-proxy primary sip-proxy.adtran.com udp 2222**

# register

Use the **register** command to define the Session Initiation Protocol (SIP) name for registration. Use the **no** form of this command to return to default settings. Variations of this command include the following:

**register** <*name*>
**register** <*name*> **auth-name** <*username*> **password** <*word*>
**register range** <*begin*> <*end*>
**register range** <*begin*> <*end*> **auth-name** <*username*> **password** <*word*>

## Syntax Description

| | |
|---|---|
| <*name*> | Specifies the desired name for the SIP trunk registration. |
| **range** <*begin*> <*end*> | Specifies the beginning and ending of the range to register. |
| **auth-name** <*username*> | Optional. Specifies the user name for authentication. |
| **password** <*word*> | Optional. Specifies the password for authentication. |

## Default Values

By default, no registration range is programmed.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example sets the register the trunk under the name of **MainOffice**:

(config)**#voice trunk T01 type sip**
(config-T01)**#register MainOffice**

# registrar expire-time <*value*>

Use the **register expire-time** command to define the Session Initiation Protocol (SIP) expiration time for registration. Use the **no** form of this command to return to default settings.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies expiration time (in seconds) for a registration. |

## Default Values

By default, no registration expiration time is programmed.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example enables registration expiration time for the SIP trunk:

(config)**#voice trunk T01 type sip**
(config-T01)**#register expire-time**

# registrar max-concurrent-reg <*value*>

Use the **registrar max-concurrent-reg** command to control the maximum number of simultaneous registrations that are allowed. This value can be adjusted to help eliminate congestion caused by too many concurrent registrations. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the maximum number of concurrent registrations. Valid range is 1 to 32 registrations. |

## Default Values

By default, the maximum number of concurrent registrations is set to 32.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example sets the maximum number on concurrent registrations to **12**:

(config)**#voice trunk T01 type sip**
(config-T01)**#registrar max-concurrent-reg 12**

# registrar primary <*value*>

Use the **registrar primary** command is used to define the primary Session Initiation Protocol (SIP) registrar fully qualified domain name (FQDN) name or IP address which is based on the DNS suffix. Use the **no** form of this command to return to default settings. Variations of this command include:

**registrar primary** <*value*>
**registrar primary** <*value*> **udp** <*number*>

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the FQDN or IP address of the registrar server. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| **udp** <*number*> | Optional. Sets the UDP port of the registrar server. Range is 0 to 65,535 ports. |

## Default Values

By default, the IP address is set to 0.0.0.0 and the UDP port is set to 5060.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Functional Notes

This command specifies which trunk will send SIP register messages. The configured value must resolve to a valid IP address.

## Usage Examples

The following example sets the registrar server to **as1.adtran.com** with a UDP port of **9060**:

(config)**#voice trunk T01 type sip**
(config-T01)**#registrar primary as1.adtran.com udp 9060**

# registrar threshold

Use the **registrar threshold** command to configure when the SIP trunk registration will be renewed. Use the **no** form of this command to return to the default value. Variations of this command include:

**registrar threshold absolute** <*value*>
**registrar threshold percentage** <*percent*>

## Syntax Description

| | |
|---|---|
| **absolute** <*time*> | Configures an absolute threshold time. This is the time in seconds between registrations. Valid range is 30 to 604,800 seconds (1 week). |
| **percentage** <*percent*> | Configures the threshold time as a percentage of the returned valid registration time. Valid range is 1 to 90 percent. |

## Default Values

By default, the registrar threshold is set at **absolute 300** seconds.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example sets the registrar threshold time at 50 percent of the returned valid registration time:

(config)**#voice trunk T01 type sip**
(config-T01)**#registrar threshold percentage 50**

# reject-external

Use the **reject-external** command to prevent inbound calls on the trunk from being routed back out of the same trunk. Use the **no** form of this command to return to default settings.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Functional Notes

In general, trunks are assigned to the user role, which means they terminate lines from a Telco provider. If this is the case, **reject-external** should be enabled so that inbound calls on the trunk cannot be routed back out of the same trunk. If the configuration is poor, inbound long distance calls could be routed back out the same trunk, causing the owner of the unit to be charged for long distance calls without his knowledge. For network-role trunks and SIP-based trunks, this command should be disabled to allow calls to be properly routed in the unit.

## Usage Examples

The following example activates **reject-external**:

(config)**#voice trunk T01 type sip**
(config-T01)**#reject-external**

# sip-keep-alive

Use the **sip-keep-alive** command to configure the type of keep-alive method for this Session Initiation Protocol (SIP) trunk. Keep-alive messages must be sent between SIP device and the registrar to keep the connected channel open for communication. Use the **no** form of this command to return to default settings. Variations of this command include the following:

**sip-keep-alive info**
**sip-keep-alive info** <*value*>
**sip-keep-alive options**
**sip-keep-alive options** <*value*>

## Syntax Description

| | |
|---|---|
| info | Specifies the INFO method to be used for the keep-alives on the trunk. |
| options | Specifies the OPTIONS method to be used for the keep-alives on the trunk. |
| <*value*> | Specifies the amount of time in seconds between the type of SIP keep-alive messages being sent during a call. Range is 30 to 3600 seconds. |

## Default Values

By default, there is no CODEC group configured.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example sets the keep-alive method to **info**:

(config)**#voice trunk T01 type sip**
(config-T01)**#sip-keep-alive info**

# sip-server primary <*value*>

Use the **sip-server primary** command to define the primary name/address of the Session Initiation Protocol (SIP) server to which the trunk will send call-related SIP messages. Use the **no** form of this command to return to default settings. Variations of this command include:

**sip-server primary** <*value*>
**sip-server primary** <*value*> **udp** <*number*>

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the fully qualified domain name (FQDN) or IP address of the SIP proxy server. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| **udp** <*number*> | Optional. Sets the UDP port of the SIP proxy server. Range is 0 to 65,535 ports. |

## Default Values

By default, the IP address is set to 0.0.0.0 and the UDP port is set to 5060.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example sets the SIP proxy server to **as1.adtran.com** with a UDP port of **9060**:

(config)**#voice trunk T01 type sip**
(config-T01)**#sip-server primary as1.adtran.com udp 9060**

# trust-domain

Use the **trust-domain** command to add security measures for users' identity and privacy by connecting the trunk to a trusted domain. Using the trusted domain adds another level of privacy from participating service providers. The system supports RFC 3323 and RFC 3325. Use the **no** form of this command to return to default settings. Variations of this command include the following:

**trust-domain**
**trust-domain p-asserted-identity-required**

## Syntax Description

| | |
|---|---|
| **p-asserted-identity-required** | Requires the use of P-Asserted-Identity SIP privacy for this trusted domain. |

## Default Values

By default, the **trust-domain** is disabled.

## Command History

| | |
|---|---|
| Release 13.1 | Command was introduced. |

## Usage Examples

The following example enables the **trust-domain**:

(config)**#voice trunk T01 type sip**
(config-T01)**#trust domain**

# VOICE TRUNK T1 COMMAND SET

To activate the Voice Trunk T1 Interface Configuration mode, enter the following command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#voice trunk t01
(config-T01)#
```

To activate the Voice Trunk T1 Feature Group D (FGD) Interface Configuration mode, enter the following command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#voice trunk t01 type t1-rbs supervision fgd role user
(config-T01)#
```

To activate the Voice Trunk Ground Start (GS) User Interface Configuration mode, enter the following command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#voice trunk t01 type t1-rbs supervision ground-start role user
(config-T01)#
```

To activate the Voice Trunk Loop Start (LS) User Interface Configuration mode, enter the following command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#voice trunk t01 type t1-rbs supervision loop-start role user
(config-T01)#
```

To activate the Voice Trunk T1 Tie Trunk (TIE) Feature Group D (FGD) Interface Configuration mode, enter the following command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#voice trunk t01 type t1-rbs supervision tie-fgd
(config-T01)#
```

To activate the Voice Trunk T1 Wink Interface Configuration mode, enter the following command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#voice trunk t01 type t1-rbs supervision wink role [network | user]
(config-T01)#
```

To activate the Voice Trunk T1 Immediate Interface Configuration mode, enter the following command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#voice trunk t01 type t1-rbs supervision immediate role [network | user]
(config-T01)#
```


*Not all Trunk T1 commands apply to all T1 trunk types. Use the ? command to display a list of valid commands.*

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:

*cross-connect on page 29*
*description <text> on page 32*
*do on page 33*
*exit on page 35*

All other commands for this command set are described in this section in alphabetical order.

*blind-dial on page 1898*
*busy all on page 1899*
*busy t1 <slot/port> tdm-group <number> on page 1900*
*caller-id on page 1901*
*caller-id-override number <number> on page 1902*
*codec-group <name> on page 1903*
*connect t1 <slot/port> tdm-group <number> on page 1904*
*did digits-transferred on page 1905*
*dnis-digits <value> prefix <number> on page 1906*
*echo-cancellation on page 1907*
*match <number> substitute <number> on page 1908*
*modem-passthrough on page 1909*
*plc on page 1910*

# blind-dial

Use the **blind dial** command to allow calls to be placed without the presence of dial-tone. Use the **no** form of this command to disable blind dialing.

## Syntax Description

No subcommands.

## Default Values

By default, **blind-dial** is disabled.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Usage Examples

The following example enables blind dialing:

(config)**#voice trunk t01 type analog supervision loop-start**
(config-t01)**#blind-dial**

# busy all

Use the **busy all** command to set all DS0s to busy so that no calls are allowed inbound or outbound. If any calls are active at the time this command is issued, the calls will stay active until either party terminates the call. Once terminated, the DS0s are busied out. Use the **no** form of this command to disable this feature. Variations of this command include:

**busy all**
**busy all now**

## Syntax Description

| | |
|---|---|
| **now** | Optional. Immediately terminates calls that are active at the time the command is issued. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example sets all DS0s on trunk T01 to busy and terminates calls that are active at the time the command is issued:

(config)**#voice trunk t01 type t1-rbs supervision wink role network**
(config-T01)**#busy all now**

# **busy t1** <*slot/port*> **tdm-group** <*number*>

Use the **busy t1 tdm-group** command to set a particular set of DS0s (defined in a TDM group) to busy so that no calls are allowed inbound or outbound the interface. If any calls are active at the time this command is issued, the calls will stay active until either party terminates the call. Once terminated, the DS0s are set to busy. Use the **no** form of this command to disable this feature. Variations of this command include:

**busy t1** <*slot/port*> **tdm-group** <*number*>
**busy t1** <*slot/port*> **tdm-group** <*number*> **now**

## Syntax Description

| | |
|---|---|
| <*slot/port*> | Specifies the slot/port for the T1. |
| <*number*> | Specifies the TDM group ID number. |
| **now** | Optional. Terminates calls that are active at the time the command is issued (for example, in the middle of a conservation). |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example sets DS0s in TDM group 2 to busy and terminates calls that are active when the command is issued:

(config)**#voice trunk t01 type t1-rbs supervision wink role network**
(config-T01)**#busy t1 0/1 tdm-group 2 now**

# caller-id

Use the **caller-id** number command to interpret and pass caller identification (ID) on this trunk. This information usually displays the name, number, time and date of the calling party. Use the **no** form of this command to cancel the setting.

## Syntax Description

No subcommands.

## Default Values

By default, caller ID is disabled.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example enables caller ID:

(config)**#voice trunk t01 type t1-rbs supervision fgd role user**
(config-T01)**#caller-id**

# caller-id-override number <*number*>

Use the **caller-id-override number** command to replace the calling party information for this trunk with a specific number. This command is used to conceal user's name and number or to display a different name and number for internal or external caller ID. Use the **no** form of this command to cancel the setting.Variations of this command include:

**caller-id-override number** <*number*>
**caller-id-override number** <*number*> <*trunk id*>

## Syntax Description

| | |
|---|---|
| <*number*> | Specifies the number to display on caller ID. |
| <*trunk id*> | Specifies the trunk id (Txx) for outbound calls. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example sets the caller ID override number on the trunk where the command is issued:

(config)**#voice trunk t01 type t1-rbs supervision wink role network**
(config-T01)**#caller-id-override number 555-8000**

# codec-group *<name>*

Use the **codec-group** command to specify the CODEC group to use for this trunk. For information on defining CODEC groups, refer to *codec-group <name>* on page 1927. Use the **no** form of this command to return to the default settings.

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies the name of the CODEC group you want to use. |

## Default Values

By default, no CODEC groups are assigned to this interface.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example associates the CODEC group named **user** with this trunk:

(config)**#voice trunk t01 type t1-rbs supervision wink role network**
(config-T01)**#codec-group user**

# connect t1 <*slot/port*> tdm-group <*number*>

Use the **connect t1 tdm-group** command to specify the physical interfaces this trunk group will use for voice connections. Refer to *tdm-group <group number>* on page 997 for more information on creating TDM groups. Use the **no** form of this command to remove physical interfaces from this trunk.

## Syntax Description

| | |
|---|---|
| <*slot/port*> | Specifies the slot/port for the T1. |
| <*number*> | Specifies the TDM group ID number. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example specifies that this trunk will use the DS0s in TDM group **3** (on T1 interface 0/1):

(config)**#voice trunk t01 type t1-rbs supervision wink role network**
(config-T01)**#connect t1 0/1 tdm-group 3**

# did digits-transferred

Use the **did digits-transferred** command to define how many of the received digits should be sent to the internal switchboard from an incoming call on a User Role Trunk. The number of digits transferred are the least digits received. Direct Inward Dialing (DID) should be used if a Telco provider is providing digits to the unit on inbound calls or if the unit needs to provide DID information to a piece of CPE equipment. Use the **no** form of this command to disable this feature. Variations of this command include:

**did digits-transferred** <*value*>
**did digits-transferred** <*value*> **prefix** <*number*>

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the number of digits to be transferred. Range is 1 to 16 digits. |
| **prefix** <*number*> | Optional. Specifies a sequence of digits to be prepended to the digits that will be transmitted. For example, if seven digits will be transferred via DID, then prefix the seven digits with 256. Thus 555-8000 would be prefixed with 256, transmitting out the string of digits 256-555-8000. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Functional Notes

User Role Example:

555-1000 is an incoming call on the trunk. With **did** set to 4, the number 1000 should be sent to the switchboard. On a network role trunk, the **did** command allows you to define how many of the digits from the Accept criteria should be sent externally from a call that was routed by the switchboard. The number of digits transferred are the least significant digits received.

Network Role Example:

555-1000 is accepted on the UT interface. With DID Digits Transferred set to 4, the number of 1000 will be sent to the device connected to the UT interface. This command cannot be specified if and when **trunk-number** is being used. Conversely, if **did** is used, **trunk-number** will not be allowed.

## Usage Examples

The following example transfers the digits 555-8000 and adds the prefix 256:

(config)**#voice trunk t01 type t1-rbs supervision wink role network**
(config-T01)**#did digits-transferred 5558000 prefix 256**

# dnis-digits <*value*> prefix <*number*>

Use the **dnis-digits prefix** command to program the number of digits to be transferred inbound on the specific trunk. Use the **no** form of this command to cancel the setting.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the number of digits to be transferred. |
| <*number*> | Specifies the number prefix to prepend to the transferred digits. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example sets the number of transferred DNIS digits to **4** on trunk T03 and sets the prefix **555**:

(config)**#voice trunk t03 type t1-rbs supervision fgd role user**
(config-T03)**#dnis-digits 4 prefix 555**

# echo-cancellation

Use the **echo-cancellation** command to improve voice quality for packetized-based voice calls such as voice over IP (VoIP) or Media Gateway Control Protocol (MGCP). Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, **echo-cancellation** is enabled.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example activates **echo-cancellation**:

(config)**#voice trunk t01 type t1-rbs supervision wink role network**
(config-T01)**#echo-cancellation**

# match <*number*> substitute <*number*>

Use the **match substitute** command to substitute a different number for the number originally dialed by a user of the system. If no match occurs (or no match statements have been entered) the original dialed number will be propagated without being modified. Use the **no** form of this command to delete match substitutions.

*You may enter multiple match commands on each trunk. The first valid match that is found for outbound numbers will be used in configurations where more than one match statement is valid for a given dialed number. Therefore, order of input is important.*

## Syntax Description

| | |
|---|---|
| **match** <*number*> | Specifies the dialed number that you are trying to match. |
| **substitute** <*number*> | Specifies the number that will be sent in place of the number that was matched. |

## Default Values

By default, no substitutions are defined.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example attempts to match the dialed number 555-8000 and specifies that number 555-8500 will be sent it no match occurs:

(config)**#voice trunk t01 type t1-rbs supervision wink role network**
(config-T01)**#match 5558000 substitute 5558500**

# modem-passthrough

Use the **modem-passthrough** command to switch to pass-through mode on fax or modem tone detection. This commands allows modem and fax calls to maintain a connection without altering the signals with the voice improvement settings such as echo cancellation and voice activity detection (vad). Use the **no** form of this command to disable this feature. Variations of this command include:

**modem-passthrough**
**modem-passthrough detection-time** <*value*>

## Syntax Description

| | |
|---|---|
| **detection-time** <*value*> | Optional. Specifies the fax and/or modem detection time length value in seconds. Range is 0 to 8 seconds. |

## Default Values

By default, **modem-passthrough** is enabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |
| Release 12.1 | Command was expanded. |

## Usage Examples

The following example disables **modem-passthrough**:

(config)**#voice trunk t01 type t1-rbs supervision wink role network**
(config-T01)**#no modem-passthrough**

# plc

Use the **plc** command to enable packet loss concealment (PLC). PLC is used to prevent choppy connections by concealing a packet loss by replacing the lost packet with another voice packet in the data stream. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, PLC is enabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example disables PLC on trunk T01:

(config)**#voice trunk t01 type t1-rbs supervision wink role network**
(config-T01)**#no plc**

# reject-external

Use the **reject-external** command to prevent inbound calls on the trunk from being routed back out of the same trunk. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, **reject-external** is enabled on this interface.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Functional Notes

In general, trunks are assigned to the user role, which means they terminate lines from a Telco provider. If this is the case, **reject-external** should be enabled so that inbound calls on the trunk cannot be routed back out of the same trunk. If the configuration is poor, inbound long distance calls could be routed back out the same trunk, causing the owner of the unit to be charged for long distance calls without his knowledge. For network-role trunks and SIP-based trunks, this command should be disabled to allow calls to be properly routed in the unit.

## Usage Examples

The following example disables **reject-external**:

(config)**#voice trunk t01 type t1-rbs supervision wink role network**
(config-T01)**#no reject-external**

# resource-selection

Use the **resource-selection** command to determine how the switchboard uses outbound call resources contained within a TDM-based trunk group. Use the **no** form of this command to disable this feature. Variations of this command include:

**resource-selection circular**
**resource-selection circular ascending**
**resource-selection circular descending**
**resource-selection linear**
**resource-selection linear ascending**
**resource-selection linear descending**

## Syntax Description

| | |
|---|---|
| **circular** | Performs call load balancing among available DS0s/B-channels in this trunk. Subsequent calls will be delivered to the next available DS0/B-channel in a round-robin fashion. |
| **linear** | Specifies that a call being delivered to this trunk will be accepted out the first available DS0/B-channel available at the time the call is received. |
| **ascending** | Optional. Distributes calls in an order from the lowest to the highest channel (DS0 1, 2, 3 through 24). |
| **descending** | Optional. Distributes calls in an order from the highest to the lowest channel (DS0 24, 23, 22 through 1). |

## Default Values

By default, resource selection is set to **linear**.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |
| Release 13.1 | Command was expanded to include the ascending and descending subcommands. |

## Usage Examples

The following example specifies circular resource selection:

(config)**#voice trunk t01 type t1-rbs supervision wink role network**
(config-T01)**#resource-selection circular**

# rtp delay-mode

Use the **rtp delay-mode** command to configure the Realtime Transport Protocol (RTP) jitter buffer packet delay mode settings. The selected setting will control how the network will handle late packets. Use the **no** form of this command to return to the default settings. Variations of this command include:

**rtp delay-mode adaptive**
**rtp delay-mode fixed**

## Syntax Description

| | |
|---|---|
| **adaptive** | Configures the RTP jitter buffer packet delay to adjust during a call based on network conditions. |
| **fixed** | Configures the RTP jitter buffer packet delay to remain constant. |

## Default Values

By default, the RTP delay mode is set to **adaptive**.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example configures RTP delay mode as fixed:

(config)**#voice trunk t01 type t1-rbs supervision wink role network**
(config-T01)**#rtp delay-mode fixed**

# rtp dtmf-relay

Use the **rtp dtmf-relay** command to configure Realtime Transport Protocol (RTP) Dual Tone Multi-Frequency (DTMF) events are relayed. The dial digits can be sent inband or out-of-band of the voice stream. Use the **no** form of this command to return to the default value. Variations of this command include:

**rtp dtmf-relay inband**
**rtp dtmf-relay nte** <*value*>

## Syntax Description

| | |
|---|---|
| **inband** | Specifies that RTP DTMF events be relayed inband in the RTP stream. |
| **nte** <*value*> | Specifies that RTP DTMF events be relayed out-of-band using NTE. Enter an NTE value between 96 and 127. |

## Default Values

By default, the **rtp dtmf-relay** is set for NTE 101.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example configures RTP DTMF relay events for **inband**:

(config)**#voice trunk T01 type t1-rbs supervision wink role network**
(config-T01)**#rtp dtmf-relay inband**

# rtp frame-packetization <*value*>

Use the **rtp frame-packetization** command to configure the Realtime Transport Protocol (RTP) frame packetization time in milliseconds for individual trunks and users. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Configures the RTP frame packetization time value in milliseconds. Select from 10, 20, or 30 milliseconds. |

## Default Values

By default, the **rtp frame-packetization** time is set to 20 milliseconds on all trunks and users.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example sets the frame packetization time for trunk T01 to 10 milliseconds:

(config)**#voice trunk T01 type t1-rbs supervision wink role network**
(config-T01)**#rtp frame-packetization 10**

# rtp packet-delay

Use the **rtp packet-delay** command to configure the maximum Realtime Transport Protocol (RTP) packet delays. This command is used to set the allowable limits of latency on the network. Use the **no** form of this command to return to default values. Variations of this command include:

**rtp packet-delay fax** <*value*>
**rtp packet-delay maximum** <*value*>
**rtp packet-delay nominal** <*value*>

## Syntax Description

| | |
|---|---|
| **fax** <*value*> | Sets the fax delay time value. Range is 0 to 500 milliseconds. |
| **maximum** <*value*> | Sets the maximum delay time value in increments of 10 milliseconds. Range is 40 to 320 milliseconds. |
| **nominal** <*value*> | Sets the nominal delay time value in increments of 10 milliseconds. Range is 10 to 240 milliseconds. |

## Default Values

By default, the RTP packet delay for fax is 300, maximum is 100, and nominal is 50.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example configures the RTP fax delay time on trunk T01 to **200** milliseconds:

(config)**#voice trunk T01 type t1-rbs supervision wink role network**
(config-T01)**#rtp packet-delay fax 200**

# rtp qos dscp <*value*>

Use the **rtp qos dscp** command to configure the maximum Realtime Transport Protocol (RTP) quality of service (QoS) parameters for differentiated services code point (DSCP). Use the **no** form of this command to return to default values.

## Syntax Description

| | |
|---|---|
| <*value*> | Configures the RTP QoS parameter for DSCP. Enter a value between 10 and 63. |

## Default Values

By default, no RTP QoS DSCP is configured for this interface.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Functional Note

By setting the **rtp qos dscp** value on an individual trunk or user, you will override the global **rtp qos dscp** setting for RTP packets. QoS is set using a Differentiated Services Code Point, or DSCP value. Valid DSCP values are 10-63, and a higher DSCP value has a higher priority. The default DSCP value for RTP is 46. Remember that if you are using a public IP connection, such as the Internet, for voice over IP, end-to-end QoS may not be guaranteed. The default DSCP value for SIP is 26. To configure QoS for the RTP traffic that carries the voice conversation, use the command, **ip rtp qos dscp** followed by the desired DSCP value.

## Usage Examples

The following example configures the RTP QoS DSCP for trunk T01 to **46**:

(config)**#voice trunk T01 type t1-rbs supervision wink role network**
(config-T01)**#rtp qos dscp 46**

# treat-inbound-as-internal

Use the **treat-inbound-as-internal** command to make incoming trunk calls appear as internal calls. Use the **no** form of this command to cancel the setting.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example configures the unit to treat inbound calls on trunk T03 as internal calls:

(config)**#voice trunk t03 type t1-rbs supervision tie-fgd role user**
(config-T03)**#treat-inbound-as-internal**

# trunk-number <*number*>

Use the **trunk number** command to define the call routing when DID is disabled. This feature directs incoming calls to the specified number when DID is not present. Use the **no** form of this command to disable this feature.

## Syntax Description

| | |
|---|---|
| <*number*> | Specify the number used for call routing when DID is disabled. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example defines call routing on trunk T01:

(config)**#voice trunk t01 type analog supervision loop-start**
(config)**#trunk-number 4000**

# tx-ani

Use the **tx-ani** command to transmit Automatic Number Identification (ANI) (calling-party number) and Dialed Number Identification Service (DNIS) (called-party number) for outbound Feature Group D (FGD) calls. This command is only valid on a trunk configured for FGD supervision. Use the **no** form of this command to cancel the setting.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example configures the system to transmit ANI/DNIS information on the outbound FGD trunk:

(config)**#voice trunk t03 type t1-rbs supervision fgd role user**
(config-T03)**#tx-ani**

# vad

Use the **vad** command to enable voice activity detection (VAD). VAD blocks out noise categorized as silence during a voice connection. The silent voice packets are not transmitted, allowing bandwidth usage to be reduced. Although VAD saves bandwidth, the quality of the voice call may be compromised. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, VAD is enabled for all T1 RBS trunks and users.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example disables VAD on trunk T01:

(config)**#voice trunk T01 type t1-rbs supervision wink role network**
(config-T01)**#no vad**

# VOICE USER CONFIGURATION COMMAND SET

To activate the Voice User Configuration mode, enter the **voice user** command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#voice user 4444
(config-4444)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:



All other commands for this command set are described in this section in alphabetical order.



# block-caller-id

Use the **block-caller-id** command to block all inbound caller ID delivery to this user. This command prevents the selected user from receiving caller ID information. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example activates the **block-called-id** command for user **4444**:

(config)**#voice user 4444**
(config-4444)**#block-caller-id**

# caller-id-override

Use the **caller-id-override** command to manipulate caller ID information for the user. This command is used to conceal user's name and number or to display a different name and number for internal or external caller ID. Use the **no** form of this command to disable this feature. Variations of this command include:

**caller-id-override external number** <*number*>
**caller-id-override internal-name empty**
**caller-id-override internal-name** <*name*>
**caller-id-override internal-number empty**
**caller-id-override internal-number** <*number*>

## Syntax Description

| | |
|---|---|
| **external-number** <*number*> | Replaces the caller ID number on external calls with the specified number. |
| **internal-name** | Replaces the caller ID name on internal calls. |
| <*name*> | Inserts this name for caller ID for internal calls. |
| **internal-number** | Replaces caller ID number on internal calls. |
| <*number*> | Inserts this number for caller ID for internal calls. |
| **empty** | Makes the caller ID name or number on internal calls display as blank. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example activates the **caller-id-override** command for external numbers for user **4444**:

(config)**#voice user 4444**
(config-4444)**#caller-id-override external-number 256-555-8000**

This examples actives the **caller-id-override** command for names with internal calls and makes the display appear blank:

(config)**#voice user 4444**
(config-4444)**#caller-id-override internal-number empty**

# call-waiting

Use the **call waiting** command to enable call waiting for a user. The selected user will be allowed to receive caller ID information. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example activates the **call-waiting** command for user **4444**:

```
(config)#voice user 4444
(config-4444)#call-waiting
```

# codec-group *<name>*

Use the **codec-group** command to configure the CODEC groups to use for the account. Generally you will have three CODEC groups: trunk, user, and fax. You will need to define CODEC lists to specify CODEC order (refer to *voice codec-list <name>* on page 700 for information). You will then use the *Voice CODEC List Configuration Command Set* on page 1747 to define the specific CODEC you want each group to use for negotiation. Use the **no** form of this command to return to the default setting.

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies the name of the CODEC group to be used for the account. Generally three basic CODEC groups are needed: **user**, **trunk**, and **fax**. |

## Default Values

By default, no CODEC groups are assigned to this interface.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example activates the CODEC group with the name **trunk**:

(config)**#voice user 4444**
(config-4444)**#codec-group trunk**

# connect

Use the **connect** command to associate physical ports with the user. This commands assigns a specific station or port type to the user. Use the **no** form of this command to return to remove associations. Variations of this command include:

**connect fxs** <*slot/port*>
**connect sip**

## Syntax Description

| | |
|---|---|
| **fxs** <*slot/port*> | Specifies that an FXS port is associated with the user. |
| **sip** | Specifies that this is a SIP user. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example associates the physical FXS slot 1/port 1 interface with the user **4444**:

(config)**#voice user 4444**
(config-4444)**#connect fxs 1/1**

The following example associates a SIP port with the user **4444.**

(config)**#voice user 4444**
(config-4444)**#connect sip**

## cos *<name>*

Use the **cos** command to set class-of-service (CoS) mode for the user. This command defines the types of phone service that will be available to the user during the day time period. Use the **no** form of this command to disable this feature.

### Syntax Description

| | |
|---|---|
| *<name>* | Specifies the predefined CoS to set for this mode. |

### Default Values

No default value necessary for this command.

### Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

### Usage Examples

The following example assigns class of service **adtran** for user **4444**:

(config)**#voice user 4444**
(config-4444)**#cos adtran**

# coverage

Use the **coverage** command to configure call coverage parameters for the user. The call coverage features automatically handles calls after a specified number of rings to user without an answer. Use the **no** form of this command to return to disable this feature. Variations of this command include:

**coverage aa** <*number*>
**coverage external** <*number*> **num-ring** <*rings*>
**coverage global** <name>
**coverage internal** <*number*> **num-ring** <*rings*>
**coverage operator num-ring** <*rings*>
**coverage vm**

## Syntax Description

| | |
|---|---|
| **aa** <*number*> | Forwards the phone to the auto attendant. It is optional to enter a specific extension programmed for the auto attendant. |
| **external** <*number*> | Forwards the phone to an external number. |
| **global** <*name*> | Uses the specified global call coverage list. |
| **internal** <*number*> | Forwards the phone to an internal number. |
| **num-ring** <*rings*> | Optional. Select the number of rings for the external number before performing the next action. Valid range is 1 to 9. |
| **operator** | Forwards the phone to the operator. |
| **vm** | Forwards the phone to voice mail. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |
| Release 11.1 | Command was updated the include the voicemail and number of rings options. |
| Release 12.1 | Command was updated the include auto attendant, global, and operator options. |

## Usage Examples

The following example specifies that the user's phone be forwarded to the internal extension **8500**.

(config)**#voice user 4444**
(config-4444)**#coverage internal 8500 num-ring 3**

# **did** <*number*>

Use the **did** command to configure direct inward dial (DID) parameters for the extension. DID is used if a service provider is providing digits to the unit on inbound calls or if the unit needs to provide DID information to a piece of CPE equipment. Use the **no** form of this command to disable this feature.

## Syntax Description

| | |
|---|---|
| <*number*> | Specifies the DID number for the user. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example assigns a **did** number of 555-4560 to user **4444**:

(config)**#voice user 4444**
(config-4444)**#did 5554000**

# directory-include

Use the **directory-include** command to add the user in a dial-by-name directory. Adding users to the directory allows the users to call parties by the voice user's name stored in the system. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example adds the user **4444** to a dial-by-name directory:

(config)**#voice user 4444**
(config-4444)**#directory-include**

# dnd

Use the **dnd** command to enable the do not disturb (DND) option for the user. This setting prevents the phone extension assigned to the user from ringing. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example enables DND for user **4444**:

(config)**#voice user 4444**
(config-4444)**#dnd**

# door-phone

Use the **door-phone** command to enable the door phone mode for this extension. Use the **no** form of this command to disable the door phone mode for this extension.

## Syntax Description

No subcommands.

## Default Values

By default, the door phone mode is disabled.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example makes extension **4444** the door phone extension.

(config)**#voice user 4444**
(config-4444)**#door-phone**

# echo-cancellation

Use the **echo-cancellation** command to improve voice quality for packetized-based voice calls such as voice over IP (VoIP) or Media Gateway Control Protocol (MGCP). Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, **echo-cancellation** is enabled.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example activates **echo-cancellation** for user **4444**:

(config)**#voice user 4444**
(config-4444)**#echo-cancellation**

# **email** <*address*>

Use the **email** command to configure the primary email notification address for this extension. Use the **no** form of this command to remove a configured email address.

## Syntax Description

| | |
|---|---|
| <*address*> | Specifies the primary email notification address for this extension. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example configures the primary email notification address for this extension as **first.last@company.com**.

(config)**#voice user 4444**
(config-4444)**#email first.last@company.com**

# email-secondary *<address>*

Use the **email-secondary** command to configure the secondary email notification address for this extension. The secondary email address will be used based on the email notification schedule. Refer to *Voice Mail Notify Schedule Command Set* on page 1794 for more information. Use the **no** form of this command to remove a configured secondary email address.

## Syntax Description

| | |
|---|---|
| *<address>* | Specifies the secondary email notification address for this extension. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example configures the secondary email notification address for this extension as **first.last@company.com**.

(config)**#voice user 4444**
(config-4444)**#email-secondary first.last@company.com**

# first-name <*name*>

Use the **first-name** command to specify the user's first name. The name will represent the user's first name in the system. Use the **no** form of this command to remove a first name.

## Syntax Description

| | |
|---|---|
| <*name*> | Specifies the user's first name. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example specifies the first name of user **4444**:

(config)**#voice user 4444**
(config-4444)**#first-name John**

# **forward** <*number*>

Use the **forward** command to redirect all calls to this extension to a specified number. Forwarding calls allows the use to receive incoming calls at a different number. Use the **no** form of this command to disable this feature.

## Syntax Description

| | |
|---|---|
| <*number*> | Forwards all calls to the specified number. Do not include dashes or hyphens in the number. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example forwards all calls to user **4444** to the number 256-555-8000:

(config)**#voice user 4444**
(config-4444)**#forward 2565558000**

# forward-disconnect

Use the **forward-disconnect** command to specify the method of manipulating the polarity to signal a disconnect of the line. Use the **no** form of this command to disable this feature. Variations of this command include:

**forward-disconnect battery remove**
**forward-disconnect battery reverse**
**forward-disconnect delay** <*value*>

## Syntax Description

| | |
|---|---|
| **battery remove** | Specifies that the battery will be removed upon disconnect. |
| **battery reverse** | Specifies that the battery will be reversed upon disconnect. |
| **delay** <*value*> | Sets a forward disconnect delay time value in milliseconds. Specify 250, 500, 750, 1000, or 2000 milliseconds. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example removes the battery upon disconnect for user **4444**. This command is used most often with a fax machine that needs to be alerted that a call has ended:

(config)**#voice user 4444**
(config-4444)**#forward-disconnect battery remove**

# fwd-courtesy

Use the **fwd-courtesy** command to enable the courtesy ring feature when a call is forwarded to notify the user that an incoming call has been re-routed. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example sends a courtesy ring when a call is forwarded:

(config)**#voice user 4444**
(config-4444)**#fwd-courtesy**

# hotel

Use the **hotel** command to allow extension reassignment to an alternate phone. Use the **no** form of this command to deny extension reassignment.

## Syntax Description

No subcommands.

## Default Values

By default, the hotel feature is disabled.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example enables the user to assign extension **4444** to an alternate phone:

```
(config)#voice user 4444
(config-4444)#hotel
```

# **hotline** <*number*>

Use the **hotline** command to configure the user's phone as a hotline phone. When the user picks up the phone, it will automatically ring the extension assigned. Use the **no** form of this command to disable this feature.

## Syntax Description

| | |
|---|---|
| <*number*> | Specifies the hotline number to dial when the phone is off-hook. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example sets up user **4444** as a hotline and specifies that the number 256-555-8000 will be dialed when user 4444's phone is off-hook:

(config)**#voice user 4444**
(config-4444)**#hotline 2565558000**

# last-name *<name>*

Use the **last-name** command to specify the user's last name. The stored name will appear in the caller ID information and directory for the user. Use the **no** form of this command to remove a last name.

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies the user's last name. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example specifies the last name of user **4444**:

(config)**#voice user 4444**
(config-4444)**#last-name Smith**

# message-waiting

Use the **message-waiting** command to configure message waiting notification methods. Use this command to select the phone alert used to notify users of new voicemail. Use the **no** form of this command to return to disable these features. Variations of this command include:

**message-waiting both**
**message-waiting dialtone-only**
**message-waiting lamp-only**

## Syntax Description

| | |
|---|---|
| **both** | Sets message-waiting notification for both dialtone and lamp. |
| **dialtone-only** | Sets message-waiting notification for dialtone only. |
| **lamp-only** | Sets message-waiting notification or lamp-only. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example sets message-waiting notification for user **4444** for both dialtone and lamp:

(config)**#voice user 4444**
(config-4444)**#message-waiting both**

# modem-passthrough

Use the **modem-passthrough** command to switch to pass-through mode on fax or modem tone detection. This commands allows modem and fax calls to maintain a connection without altering the signals with the voice improvement settings such as echo cancellation and voice activity detection (vad). Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, **modem-passthrough** is enabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example disables modem-passthrough:

(config)**#voice user 4444**
(config-4444)**#no modem-passthrough**

# num-rings <*value*>

Use the **num-rings** command to specify the number of rings per station before the system redirects the call. Use the **no** form of this command to return to disable this feature.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the number of rings before the next action. Specify 0 trough 9 rings. Entering 0 specifies an unlimited number of rings. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following sets the password for user **4444** to **4321**:

(config)**#voice user 4444**
(config-4444)**#num-rings 6**

# password *<password>*

Use the **password** command to create a password or personal identification number (PIN) number to protect voice settings and messages. Use the **no** form of this command to remove a password.

## Syntax Description

| | |
|---|---|
| *<password>* | Specifies a 4-digit password (or PIN) number. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following sets the password for user **4444** to **4321**:

(config)**#voice user 4444**
(config-4444)**#password 4321**

# phone mac-address *<mac address>*

Use the **phone mac-address** command to configure the user's SIP phone MAC address (in hexadecimal notation). Use the **no** form of this command to remove the SIP phone MAC address information for this user.

## Syntax Description

| | |
|---|---|
| *<mac address>* | Specifies a SIP phone MAC address. MAC addresses should be expressed in following format xx:xx:xx:xx:xx:xx (for example, **00:A0:C8:00:00:01**). |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example sets the user's phone MAC address to **00:0A:C8:5F:00:D2**:

(config)**#voice user 4444**
(config-4444)**#phone mac-address 00:0A:C8:5F:00:D2**

# phone model <*value*>

Use the **phone model** command to configure the user's SIP phone model. This information is used when generating a configuration file for the user's account. Use the **no** form of this command to remove the SIP phone model information for this user.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies a SIP phone model. Type **phone model ?** for a complete list of available phones. |

## Default Values

By default, the phone model is set to Polycom 50x.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example set the user's SIP phone model to Polycom Soundpoint IP 601:

(config)**#voice user 4444**
(config-4444)**#phone model polycom 601**

# plc

Use the **plc** command to enable packet loss concealment (PLC). PLC is used to prevent choppy connections by concealing a packet loss by replacing the lost packet with another voice packet in the data stream. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, PLC is enabled.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following disables PLC the for user **4444**:

(config)**#voice user 4444**
(config-4444)**#no plc**

# rtp delay-mode

Use the **rtp delay-mode** command to configure the Realtime Transport Protocol (RTP) jitter buffer packet delay mode settings. RTP is used to prevent static voice connections by enhancing the quality of the packet delivery. Use the **no** form of this command to return to the default setting. Variations of this command include:

**rtp delay-mode adaptive**
**rtp delay-mode fixed**

## Syntax Description

| | |
|---|---|
| **adaptive** | Configures RTP jitter buffer packet delay to adjusts during a call based on network conditions. |
| **fixed** | Configures RTP jitter buffer packet delay remains constant. |

## Default Values

By default, the RTP delay mode is set to **adaptive**. This allows for minimal latency by adjusting the average packet delay based on the conditions of the network.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example configures RTP delay mode as fixed:

(config)**#voice user 4444**
(config-4444)**#rtp delay-mode fixed**

# rtp dtmf-relay

Use the **rtp dtmf-relay** command to configure Realtime Transport Protocol (RTP) Dual Tone Multi-Frequency (DTMF) events are relayed. The dial digits can be sent inband or out-of-band of the voice stream. Use the **no** form of this command to return to the default value. Variations of this command include:

**rtp dtmf-relay inband**
**rtp dtmf-relay nte** <*value*>

## Syntax Description

| | |
|---|---|
| **inband** | Specifies that RTP DTMF events be relayed inband in the RTP stream. |
| **nte** <*value*> | Specifies that RTP DTMF event value be relayed out-of-band using NTE. Enter an NTE value between 96 and 127. |

## Default Values

By default, the **rtp dtmf-relay** is set for NTE 101.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example configures RTP DTMF relay events for **inband**:

(config)**#voice user 4444**
(config-4444)**#rtp dtmf-relay inband**

# rtp frame-packetization <*value*>

Use the **rtp frame-packetization** command to configure the Realtime Transport Protocol (RTP) frame packetization time in milliseconds. Use the **no** form of this command to return to the default value.

## Syntax Description

| | |
|---|---|
| <*value*> | Configures the RTP frame packetization time value in milliseconds. Select from 10, 20, or 30 milliseconds. |

## Default Values

By default, the **rtp frame-packetization** time is set to 20 milliseconds on all trunks and users.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example sets the frame packetization time for user **4444** to **10** milliseconds:

(config)**#voice user 4444**
(config-4444)**#rtp frame-packetization 10**

# rtp packet-delay

Use the **rtp packet-delay** command to configure the maximum Realtime Transport Protocol (RTP) packet delays. This command is used to set the allowable limits of latency on the network. Use the **no** form of this command to return to default values. Variations of this command include:

**rtp packet-delay fax** <*value*>
**rtp packet-delay maximum** <*value*>
**rtp packet-delay nominal** <*value*>

## Syntax Description

| | |
|---|---|
| **fax** <*value*> | Sets the fax delay time value in milliseconds. Range is 0 to 500 milliseconds. |
| **maximum** <*value*> | Sets the maximum delay time value in increments of 10 milliseconds. Range is 40 to 320 milliseconds. |
| **nominal** <*value*> | Sets the nominal delay time value in increments of 10 milliseconds. Range is 10 to 240 milliseconds. |

## Default Values

By default, the RTP packet delay for fax is 300, maximum is 100, and nominal is 50.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example configures the RTP fax delay time for user **4444** to **200** milliseconds:

(config)**#voice user 4444**
(config-4444)**#rtp packet-delay fax 200**

# rtp qos dscp <*code*>

Use the **rtp qos dscp** command to configure the maximum Realtime Transport Protocol (RTP) quality of service (QoS) parameters for differentiated services code point (DSCP). Use the **no** form of this command to return to default values.

## Syntax Description

| | |
|---|---|
| <*code*> | Configures the RTP QoS parameter for DSCP. Enter a value between 10 and 63. |

## Default Values

By default, no RTP QoS DSCP is configured for this interface.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Functional Note

By setting the **rtp qos dscp** value on an individual trunk or user, you will override the global **rtp qos dscp** setting for RTP packets. QoS is set using a Differentiated Services Code Point, or DSCP value. Valid DSCP values are 10-63, and a higher DSCP value has a higher priority. The default DSCP value for RTP is 46. Remember that if you are using a public IP connection, such as the Internet, for voice over IP, end-to-end QoS may not be guaranteed. The default DSCP value for SIP is 26. To configure QoS for the RTP traffic that carries the voice conversation, use the command, **ip rtp qos dscp** followed by the desired DSCP value.

## Usage Examples

The following example configures the RTP QoS DSCP for user **4444** to **46**:

(config)**#voice user 4444**
(config-4444)**#rtp qos dscp 46**

# sip-identity

Use the **sip-identity** command to configure the Session Initiation Protocol (SIP) registration options for the user. Use the **no** form of the command to disable the settings. Variations of this command include the following:

**sip-identity** <*station*> <*Txx*>
**sip-identity** <*station*> <*Txx*> **register**
**sip-identity** <*station*> <*Txx*> **register auth-name** <*username*> **password** <*password*>

## Syntax Description

| | |
|---|---|
| <*station*> <*Txx*> | Specifies the station to be used for SIP trunk (e.g., station extension) and the SIP trunk (Txx, e.g. T01) through which to register the server. |
| **register** | Registers the user to the server. |
| **register auth-name** <*username*> | Optional. Sets the username that will be required as authentication for registration to the SIP server. |
| **password** <*password*> | Optional. Sets the password that will be required as authentication for registration to the SIP server. |

## Default Values

No default values necessary for this command.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example specifies trunk **T02** and extension **4400** for SIP **identity**:

(config)**#voice user 4444**
(config-4444)**#sip identity 4400 T02**

# special-ring-cadences

Use the **special-ring-cadences** command to enable special ring cadences for this user. This command allows the user to be alerted with a distinctive ring. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example configures enables special ring cadences for user **4444**:

(config)**#voice user 4444**
(config-4444)**#special-ring-cadences**

# speed-dial *<number>*

Use the **speed-dial** command to assign a number (1 though 20) to the user. The speed dial number allows the users to call each other by simply dialing 1 or 2 digit numbers. Use the **no** form of this command to disable this feature.

## Syntax Description

| | |
|---|---|
| *<number>* | Specifies the speed dial number for the user. Select from numbers 1 through 20. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example assigns a **speed-dial** number to user **4444**:

(config)**#voice user 4444**
(config-4444)**#speed-dial 2**

# station-lock

Use the **station-lock** command to lock the (station) against inbound or outbound calls. Locking a station will restrict phone privileges. Use the **no** form of this command to disable this feature. Variations of this command include:

**station-lock admin**
**station-lock admin inbound**
**station-lock admin inbound-outbound**
**station-lock user**
**station-lock user inbound**
**station-lock user inbound-outbound**

## Syntax Description

| | |
|---|---|
| **admin** | Allows the administrator to block calls. |
| **user** | Allows the user to block calls. |
| **inbound** | Blocks inbound calls. |
| **inbound-outbound** | Blocks both inbound and outbound calls. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example configures an administrator inbound and outbound station lock:

(config)**#voice user 4444**
(config-4444)**#station-lock admin inbound-outbound**

# vad

Use the **vad** command to enable voice activity detection (VAD). VAD blocks out noise categorized as silence during a voice connection. The silent voice packets are not transmitted, allowing bandwidth usage to be reduced. Although VAD saves bandwidth, the quality of the voice call may be compromised. Use the **no** form of this command to disable this feature.

## Syntax Description

No subcommands.

## Default Values

By default, VAD is enabled for all voice trunks and users.

## Command History

| | |
|---|---|
| Release 9.3 | Command was introduced. |

## Usage Examples

The following example disables VAD for user **4444**:

(config)**#voice user 4444**
(config-4444)**#no vad**

# voicemail cos *<name>*

Use the **voicemail cos** command to set the voice mail class-of-service (CoS) rule set for this user. Refer to *Voice Mail CoS Command Set* on page 1786 for more information on commands in the voice mail CoS rule set. Use the **no** form of this command to delete the CoS rule set for this user.

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies voice mail CoS rule set. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example set the voice mail CoS for this user to **class1**.

(config)**#voice user 4444**
(config-4444)**#voicemail cos class1**

# voicemail notify

Use the **voicemail notify** command to create a voice mail notification schedule for this extension. Use the **no** form of this command to delete the voice mail notification schedule. Variations of this command include:

**voicemail notify email text-only**
**voicemail notify schedule** <*day*> <*time*> **[am | pm]**

## Syntax Description

| | |
|---|---|
| **email text-only** | Specifies that notifications will be sent via email. Specify the email address to be used with the **email** <*address*> command. |
| <*day*> | Specifies the day of the week for the notification schedule. |
| <*time*> | Specifies the time of day for the notification schedule. Specify time in the format HH:MM. |
| **am \| pm** | Specifies a.m. or p.m. as the time for the notification schedule. |

## Default Values

By default, the notification schedule is set to run on Sunday at 11:59 p.m.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example sets the email notification schedule for **monday** at **6:00 am**.

(config)**#voice user 4444**
(config-4444)**#voicemail notify schedule monday 06:00 am**

# voicemail oper-assist <*number*>

Use the **voicemail oper-assist** command to configure the number to which operator calls from this extension will be forwarded. Use the **no** form of this command to return to the default values.

## Syntax Description

| | |
|---|---|
| <*number*> | Specifies the extension to which operator calls will be forwarded. |

## Default Values

By default, operator calls are forwarded to extension 0.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example forwards all operator calls from this extension to extension **1234**.

(config)**#voice user 4444**
(config-4444)**#voicemail oper-assist 1234**

# voicemail password *<password>*

Use the **voicemail password** command to create a voice mail password for this extension. Use the **no** form of this command to delete the voice mail password

## Syntax Description

| | |
|---|---|
| *<password>* | Specifies the four-digit voice mail password for this extension. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 12.1 | Command was introduced. |

## Usage Examples

The following example sets the voice mail password for this extension to **3456**.

(config)**#voice user 4444**
(config-4444)**#voicemail password 3456**

# DHCP POOL COMMAND SET

To activate the DHCP Pool mode, enter the **ip dhcp-server pool** command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#ip dhcp-server pool MyPool
(config-dhcp)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:



All other commands for this command set are described in this section in alphabetical order.



# bootfile *<name>*

Use the **bootfile** command to specify a fully qualified directory-path name to a file located on a TFTP server on the network. Some network devices use the file (the path sent to the DHCP client in the DHCPOFFER message) for initial configuration. Use the **no** form of this command to remove a configured boot file.

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies a fully qualified directory-path name to the file located on the network. If the file is located in the root directory of the TFTP server, enter the file name only. |

## Default Values

By default there is no specified boot file.

## Functional Notes

RFC2131 provides specifications for DHCP servers to supply clients with information that allows the clients to exchange packets with other hosts on the network. DHCP clients that do not store the correct boot software on an internal flash drive can receive a boot file from a TFTP server. The AOS DHCP server can provide these devices with the address of the network TFTP server and the configuration file name. For example, some IP phones use this functionality to download the feature and key activation file. Use the **tftp-server** command (on page 1982) to specify the IP address of the network TFTP server.

RFC2131 includes provisions to allow DHCP servers to utilize the 128 octets designated for the boot file directory-path for expanding the DHCP options field. RFC1533 outlines the available DHCP variables for the options field. This process must be negotiated between client and server during the DHCPDISCOVER process and should only take place if the client specifies a small maxDHCPmessage size in the DHCPDISCOVER message.

## Usage Examples

The following example specifies the location of a TFTP server on the LAN at **10.10.0.4** and a boot file of **myconfig.cfg** (located in the TFTP server root directory) for the DHCP pool **IP_Phones**:

(config)**#ip dhcp-server pool IP_Phones**
(config-dhcp)**#tftp sever 10.10.0.4**
(config-dhcp)**#bootfile myconfig.cfg**

# client-identifier *<identifier>*

Use the **client-identifier** command to specify a unique identifier (in dotted hexadecimal notation) for a Dynamic Host Configuration Protocol (DHCP) client. Use the **no** form of this command to remove a configured client identifier.

## Syntax Description

| | |
|---|---|
| *<identifier>* | Specifies a client identifier using 7 to 28 hexadecimal characters with colon delimiters. Refer to the *Functional Notes* below for more information. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |

## Functional Notes

DHCP clients use client identifiers in place of hardware addresses. To create the client-identifier, begin with the two-digit numerical code representing the media type and append the client's MAC address. For example, a Microsoft client with an Ethernet (01) MAC address d2:17:04:91:11:50 uses a client identifier of 01:d2:17:04:91:11:50.

## Usage Examples

The following example specifies the client identifier for a Microsoft client with an Ethernet MAC address of **d217.0491.1150**:

(config)**#ip dhcp-server pool Microsoft_Clients**
(config-dhcp)**#client-identifier 01:d2:17:04:91:11:50**

# client-name <*name*>

Use the **client-name** command to specify the name of a Dynamic Host Configuration Protocol (DHCP) client. Use the **no** form of this command to remove the configured client name.

## Syntax Description

| | |
|---|---|
| <*name*> | Identifies the DHCP client (example is **client1**) using an alphanumeric string (up to 32 characters in length). |

*The specified client name should not contain the domain name.*

## Default Values

By default, there are no specified client names.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |

## Usage Examples

The following example specifies a client name of **myclient**:

(config)**#ip dhcp-server pool MyPool**
(config-dhcp)**#client-name myclient**

# **default-router** *<ip address>*

Use the **default-router** command to specify the default primary and secondary routers to use for the Dynamic Host Configuration Protocol (DHCP) client. Use the **no** form of this command to remove the configured router. Variations of this command include:

**default-router** *<ip address>*
**default-router** *<ip address> <secondary>*

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies the IP address of the preferred router on the client's subnet. |
| *<secondary>* | Optional. Specifies the IP address of the second preferred router on the client's subnet.<br>IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

## Default Values

By default, there are no specified default routers.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |

## Functional Notes

When specifying a router to use as the primary/secondary preferred router, verify that the listed router is on the same subnet as the DHCP client. The AOS allows a designation for two routers, listed in order of precedence.

## Usage Examples

The following example configures a default router with address **192.22.4.253** and a secondary router with address **192.22.4.254**:

(config)**#ip dhcp-server pool MyPool**
(config-dhcp)**#default-router 192.22.4.253 192.22.4.254**

# dns-server *<ip address>*

Use the **dns-server** command to specify the default primary and secondary Domain Name System (DNS) servers to use for the Dynamic Host Configuration Protocol (DHCP) client. Use the **no** form of this command to remove the configured DNS server. Variations of this command include:

**dns-server** *<ip address>*
**dns-server** *<ip address>* *<secondary>*

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies the IP address of the preferred DNS server on the network. |
| *<secondary>* | Optional. Specifies the IP address of the second preferred DNS server on the network.<br>IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

## Default Values

By default, there are no specified default DNS servers.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |

## Usage Examples

The following example specifies a default DNS server with address **192.72.3.254** and a secondary DNS server with address **192.100.4.253**:

(config)**#ip dhcp-server pool MyPool**
(config-dhcp)**#dns-server 192.72.3.254 192.100.4.253**

# domain-name <*name*>

Use the **domain-name** command to specify the domain name for the Dynamic Host Configuration Protocol (DHCP) client. Use the **no** form of this command to remove the configured domain name.

## Syntax Description

| | |
|---|---|
| <*name*> | Identifies the DHCP client (e.g., adtran.com) using an alphanumeric string (up to 32 characters in length). |

## Default Values

By default, there are no specified domain names.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |

## Usage Examples

The following example specifies a domain name of **adtran.com**:

(config)**#ip dhcp-server pool MyPool**
(config-dhcp)**#domain-name adtran.com**

# **hardware-address** <*mac address*>

Use the **hardware-address** command to specify the name of a Dynamic Host Configuration Protocol (DHCP) client. Use the **no** form of this command to remove the configured client name. Variations of this command include:

**hardware-address** <*mac address*>
**hardware-address** <*mac address*> <*type*>
**hardware-address** <*mac address*> **ethernet**
**hardware-address** <*mac address*> **ieee802**

## Syntax Description

| | | |
|---|---|---|
| <*mac address*> | Specifies a valid 48-bit MAC address. MAC addresses should be expressed in following format xx:xx:xx:xx:xx:xx (for example, **00:A0:C8:00:00:01**). | |
| <*type*> | Optional. Specifies one of the hardware types listed in RFC1700. Valid range is 1 to 21. | |
| | The valid hardware types are as follows: | |
| | 1 | 10 Mb Ethernet |
| | 2 | Experimental 3 Mb Ethernet |
| | 3 | Amateur Radio AX.25 |
| | 4 | Proteon ProNET Token Ring |
| | 5 | Chaos |
| | 6I | EEE 802 Networks |
| | 7 | ARCNET |
| | 8 | Hyperchannel |
| | 9 | Lanstar |
| | 10 | Autonet Short Address |
| | 11 | LocalTalk |
| | 12 | LocalNet (IBM PCNet or SYTEK LocalNet) |
| | 13 | Ultra link |
| | 14 | SMDS |
| | 15 | Frame Relay |
| | 16 | Asynchronous Transmission Mode (ATM) |
| | 17 | HDLC |
| | 18 | Fibre Channel |
| | 19 | Asynchronous Transmission Mode (ATM) |
| | 20 | Serial Line |
| | 21 | Asynchronous Transmission Mode (ATM) |
| **ethernet** | Optional. Specifies standard Ethernet networks. | |
| **ieee802** | Optional. Specifies IEEE 802 standard networks. | |

## Default Values

By default, the hardware address type is set to 10 Mbps Ethernet (**1**).

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |

## Usage Examples

The following example specifies an Ethernet client with a MAC address of **ae:11:54:60:99:10**:

(config)**#ip dhcp-server pool MyPool**
(config-dhcp)**#hardware-address ae:11:54:60:99:10 Ethernet**

# **host** *<ip address>*

Use the **host** command to specify the IP address and subnet mask for a manual binding to a Dynamic Host Configuration Protocol (DHCP) client. Use the **no** form of this command to remove the configured client address. Variations of this command include:

**host** <*ip address*>
**host** <*ip address*> <*subnet mask*>

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies the IP address for a manual binding to a DHCP client. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| *<subnet mask>* | Optional. Specifies the subnet mask that corresponds to a range of IP addresses (network) or a specific host. Subnet masks can be expressed in dotted decimal notation (for example, **255.255.255.0**) or as a prefix length (for example, **/24**).<br>If the subnet mask is left unspecified, the DHCP server examines its address pools to obtain an appropriate mask. If no valid mask is found in the address pools, the DHCP server uses the Class A, B, or C natural mask. |

## Default Values

By default, there are no specified host addresses.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |

## Usage Examples

The following examples show two different ways to specify a client with IP address **12.200.5.99** and a 21-bit subnet mask:

(config)**#ip dhcp-server pool MyPool**
(config-dhcp)**#host 12.200.5.99 255.255.248.0**

*or*

(config)**#ip dhcp-server pool MyPool**
(config-dhcp)**#host 12.200.5.99/21**

# lease *<days>*

Use the **lease** command to specify the duration of the lease for an IP address assigned to a Dynamic Host Configuration Protocol (DHCP) client. Use the **no** form of this command to return to the default lease value. Variations of this command include:

**lease** *<days>*
**lease** *<days>* *<hours>*
**lease** *<days>* *<hours>* *<minutes>*

## Syntax Description

| | |
|---|---|
| *<days>* | Specifies the duration of the IP address lease in days. |
| *<hours>* | Optional. Specifies the number of hours in a lease. You may only enter a value in the hours field if the days field is specified. |
| *<minutes>* | Optional. Specifies the number of minutes in a lease. You may only enter a value in the minutes field if the days and hours fields are specified. |

## Default Values

By default, an IP address lease is one day.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |

## Usage Examples

The following example specifies a lease of **2** days:

(config)**#ip dhcp-server pool MyPool**
(config-dhcp)**#lease 2**

The following example specifies a lease of **1 hour**:

(config)**#ip dhcp-server pool MyPool**
(config-dhcp)**#lease 0 1**

The following example specifies a lease of **30 minutes**:

(config)**#ip dhcp-server pool MyPool**
(config-dhcp)**#lease 0 0 30**

# netbios-name-server *<ip address>*

Use the **netbios-name-server** command to specify the primary and secondary NetBIOS Windows Internet Naming Service (WINS) name servers available for use by the Dynamic Host Configuration Protocol (DHCP) clients. Use the **no** form of this command to remove a configured NetBIOS name server. Variations of this command include:

**netbios-name-server** *<ip address>*
**netbios-name-server** *<ip address>* *<secondary>*

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies the IP address of the preferred NetBIOS WINS name server on the network. |
| *<secondary>* | Optional. Specifies the IP address of the second preferred NetBIOS WINS name server on the network.<br>IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

## Default Values

By default, there are no configured NetBIOS WINS name servers.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |

## Usage Examples

The following example specifies a primary NetBIOS WINS name server with an IP address of **172.45.6.99** and a secondary with an IP address of **172.45.8.15**:

(config)**#ip dhcp-server pool MyPool**
(config-dhcp)**#netbios-name-server 172.45.6.99 172.45.8.15**

# netbios-node-type

Use the **netbios-node-type** command to specify the type of NetBIOS node used with Dynamic Host Configuration Protocol (DHCP) clients. Use the **no** form of this command to remove a configured NetBIOS node type. Variations of this command include:

**netbios-node-type** <*value*>
**netbios-node-type b-node**
**netbios-node-type h-node**
**netbios-node-type m-node**
**netbios-node-type p-node**

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the NetBIOS node type using the numerical value. Refer to the node types below for the corresponding numerical values. |
| **b-node** | Specifies the broadcast node. Numeric value is 1. |
| **h-node** | Specifies the hybrid node (recommended). Numeric value is 8. |
| **m-node** | Specifies the mixed node. Numeric value is 4. |
| **p-node** | Specifies the peer-to-peer node. Numeric value is 2. |

## Default Values

By default, the **netbios-node-type** is set to **h-node** (8) - Hybrid node.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |

## Usage Examples

The following example specifies a client's NetBIOS node type as **h-node**:

(config)**#ip dhcp-server pool MyPool**
(config-dhcp)**#netbios-node-type h-node**

Alternately, the following also specifies the client's NetBIOS node type as **h-node**:

(config-dhcp)**#netbios-node-type 8**

# **network** *<ip address>*

Use the **network** command to specify the subnet number and mask for an AOS Dynamic Host Configuration Protocol (DHCP) server address pool. Use the **no** form of this command to remove a configured subnet. Variations of this command include:

**network** <*ip address*>
**network** <*ip address*> <*subnet mask*>

## Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies the IP address of the DHCP address pool. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |
| *<subnet mask>* | Optional. Specifies the subnet mask that corresponds to a range of IP addresses (network) or a specific host. Subnet masks can be expressed in dotted decimal notation (for example, **255.255.255.0**) or as a prefix length (for example, **/24**). If the subnet mask is left unspecified, the DHCP server uses the Class A, B, or C natural mask. |

## Default Values

By default, there are no configured DHCP address pools.

## Command History

| | |
|---|---|
| Release 2.1 | Command was introduced. |

## Usage Examples

The following examples show two different ways to configure an address pool subnet of **192.34.0.0** with a 16-bit subnet mask:

(config)**#ip dhcp-server pool MyPool**
(config-dhcp)**#network 192.34.0.0 255.255.0.0**

*or*

(config)**#ip dhcp-server pool MyPool**
(config-dhcp)**#network 192.34.0.0 /16**

## ntp-server *<ip address>*

Use the **ntp-server** command to specify the name of the Network Time Protocol (NTP) server published to the client. Use the **no** form of this command to remove a defined NTP server.

### Syntax Description

| | |
|---|---|
| *<ip address>* | Specifies the IP address of the NTP server. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

### Default Values

By default, no NTP server is defined.

### Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

### Usage Examples

The following example specifies the IP address of the NTP server:

(config)**#ip dhcp pool MyPool**
(config-dhcp)**#ntp-server 192.168.1.1**

# option

Use the **option** command to describe a generic DHCP option to be published to the client. The user may specify any number of generic options to be published to the client. Use the **no** form of this command to return to the default value. Variations of this command include:

**option** <*option value*> **ascii** <*value*>
**option** <*option value*> **hex** <*value*>
**option** <*option value*> **ip** <*value*>

## Syntax Description

| | |
|---|---|
| <*option value*> | Specifies the value of the generic DHCP option published to the client. Range is 0 to 255. |
| **ascii** | Specifies the DHCP option information in ascii format. |
| **hex** | Specifies the DHCP option information in hexidecimal format. |
| **ip** | Specifies the DHCP option information in IP format. |
| <*value*> | Specifies the ASCII, hexidecimal, or IP value. The value for **ascii** is simple text. The value for **hex** is an 8-digit hexidecimal number (32 bit). The value for **ip** is an IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Usage Examples

The following example publishes DHCP options to the client:

(config)**#ip dhcp pool MyPool**
(config-dhcp)**#option 100 ascii ascii_value**
(config-dhcp)**#option 101 hex AB458E80**
(config-dhcp)**#option 102 ip 192.168.1.1**

# tftp-server <*name*>

Use the **tftp-server** command to specify the IP address or DNS name of the TFTP server published to the client. Use the **no** form of this command to remove a defined TFTP server.

## Syntax Description

| | |
|---|---|
| <*name*> | Specifies the DNS name or dotted notation IP address of the server. |

## Default Values

By default, no TFTP server is defined.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Usage Examples

The following example specifies the IP address of the TFTP server:

(config)**#ip dhcp pool MyPool**
(config-dhcp)**#tftp-server 192.168.1.1**

The following example specifies the DNS name of the TFTP server:

(config)**#ip dhcp pool MyPool**
(config-dhcp)**#tftp-server MyServer.adtran.com**

# timezone-offset <*value*>

Use the **timezone-offset** command to specify the timezone adjustment (in hours) published to the client. Use the **no** form of this command to remove a timezone offset.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the timezone adjustment (in hours) published to the client. Use an integer from -12 to 12 hours. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 9.1 | Command was introduced. |

## Usage Examples

The following example sets the timezone adjustment for the client to **-3** hours. For example, if the server time is configured for Eastern time and the client is configured for Pacific time, you can set the client timezone adjustment to -3 hours:

(config)**#ip dhcp pool MyPool**
(config-dhcp)**#timezone-offset -3**

# QUALITY OF SERVICE (QOS) MAP COMMAND SET

A QoS policy is defined using a QoS map in the AOS CLI. The QoS map is a named list with sequenced entries. An entry contains a single match reference and one or more actions (priority, set, or both). To activate the QoS Command Set (which allows you to create and/or edit a map), enter a valid version of the QoS command at the Global Configuration mode prompt. Multiple map entries for the same QoS map are differentiated by a sequence number. The sequence number is used to assign match order.

Once created, a QoS map must be applied to an interface (using the **qos-policy out** <*map-name*> command) in order to actively process traffic. Any traffic for the interface that is not sent to the priority queue is sent using the default queuing method for the interface (such as weighted fair queuing). For example:

```
>enable
#config terminal
(config)#qos map VOICEMAP 10
(config-qos-map)#match precedence 5
(config-qos-map)#priority 512
(config-qos-map)#exit
(config)#interface fr 1
(config-fr 1)#qos-policy out VOICEMAP
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:



All other commands for this command set are described in this section in alphabetical order.



# bandwidth

Use the **bandwidth** command to specify bandwidth allocation for individual traffic classes (for CBWFQ configurations). Use the **no** form of this command to remove a configured bandwidth allocation. Variations of this command include:

**bandwidth** <*value*>
**bandwidth percent** <*value*>
**bandwidth remaining percent** <*value*>

## Syntax Description

| | |
|---|---|
| <*value*> | Allocates the minimum bandwidth for a traffic class, specifying the minimum as an absolute bandwidth in kilobits per second. Range is 8 to 2,000,000 kbps). |
| **percent** <*value*> | Allocates a minimum bandwidth for a traffic class, specifying the minimum as a percentage of the total interface bandwidth. See *Functional Notes* for more details. |
| **remaining percent** <*value*> | Allocates a minimum bandwidth for a traffic class, specifying the minimum as a percentage of the total interface bandwidth not allocated to priority classes in the QoS map. See *Functional Notes* for more details. |

## Default Values

By default, there is no bandwidth allocation configured for a QoS map entry.

## Functional Notes

When configuring **bandwidth** allocations for CBWFQ, there are a few rules that must be obeyed.

1. The units of the bandwidth (kbps, percent, or remaining percent) must be consistent for all class-based entries (using the **bandwidth** command) in a QoS map set.

2. The total bandwidth between all **priority** entries and class-based entries (**bandwidth**) in a QoS map set should not be configured beyond the specified **max-reserved-bandwidth** (default 75 percent) on the interface that the QoS policy is applied to (using the **qos-policy** command), or the map will be disabled. Even with the configuration limit, class-based queues can still use more than the **max-reserved-bandwidth** limitation, up to 100 percent of the bandwidth, if enough traffic is present. When the configured QoS map is applied to a physical interface, AOS displays bandwidth information for the map and the physical interface. For example, the Frame Relay interface (**fr 1**) has been connected to the E1 interface (**e1 1/1**) using the **cross-connect** command. Applying the QoS map (**MyMapA**) to the Frame Relay interface (**fr 1**) produces the following status message:

   2005.08.09 07:28:22 QOS.INTERFACE QOS policy “MyMapA” requires 1288 kbps of bandwidth and 1488 kbps is now available for interface fr 1 -> the QOS policy for this port has been forced ACTIVE.

   This status message displays the sum total of the bandwidths specified in the QoS map (1288 kbps) and the available interface bandwidth using the total line rate configured on the interface (1488 kbps).

3. Up to four class-based entries (**bandwidth** commands) can be configured in a particular QoS map set. Up to 16 class-based entries can be configured (four entries on four QoS maps).
4. Within a QoS map entry, CBWFQ bandwidth and low latency priority actions are mutually exclusive. However, bandwidth and priority actions may be applied to different entries in the same QoS map.

**Determining Bandwidth Entries**

*When possible, use the **bandwidth** <value> command to specify an absolute amount of bandwidth (in kbps) for the traffic class.*

When determining the **percent** <*value*> entry, use the following formula:

$$\frac{\text{Bandwidth}}{\text{Line Rate}}$$

where

| | |
|---|---|
| *Bandwidth* | Specifies the minimum amount of bandwidth needed for the traffic (in kbps). |
| *Line Rate* | Specifies the total data rate configured on the interface (for example, 8 DS0s (64 kbps per DS0) on a T1 equals a line rate of 512 kbps). |

For example, to specify 80 kbps of data on an interface with a total of 512 kbps of available bandwidth, enter the following command:

(config-qos-map 1)**#bandwidth percent 16**

When determine the **remaining percent** <*value*> entry, use the following formula:

$$\frac{\text{Bandwidth}}{[(\text{max-reserved-bandwidth})(\text{Line Rate})] - \text{Priority Traffic}}$$

where

| | |
|---|---|
| *Bandwidth* | Specifies the minimum amount of bandwidth needed for the traffic (in kbps). |
| *max-reserved-bandwidth* | Specifies the percentage of the total line rate available for use by QoS. |
| *Line Rate* | Specifies the total data rate configured on the interface (for example, 8 DS0s (64 kbps per DS0) on a T1 equals a line rate of 512 kbps). |
| *Priority Traffic* | Specifies the amount of bandwidth reserved using the **priority** command. |

For example, to specify 80 kbps of data on an interface with a total of 512 kbps of available bandwidth, 256 kbps reserved (using the **priority** command), and reserving 15 percent of the bandwidth for routing and L2 protocol traffic (**max-reserved-bandwidth** = 85) enter the following command:

(config-qos-map 1)**#bandwidth remaining percent 45**

## Usage Examples

The following example creates a QoS map with four traffic classes (based on IP packet precedence values) and allocates bandwidth to each class:

(config)**#qos map MyMap 1**
(config-qos-map)**#match precedence 5**
(config-qos-map)**#bandwidth percent 25**
(config)**#qos map MyMap 2**
(config-qos-map)**#match precedence 3**
(config-qos-map)**#bandwidth percent 10**
(config)**#qos map MyMap 3**
(config-qos-map)**#match precedence 2**
(config-qos-map)**#bandwidth percent 10**
(config)**#qos map MyMap 4**
(config-qos-map)**#match precedence 1**
(config-qos-map)**#bandwidth percent 15**

# match

Use the **match** command to specify which traffic should be processed by this QoS map. Use the **no** forms of these commands to discontinue matching.Possible variations of this command include:

**match dscp** <*value*>
**match ip rtp** <*port #*>
**match ip rtp** <*begin port range*> <*end port range*>
**match ip rtp** <*begin port range*> <*end port range*> **all**
**match list** <*name*>
**match precedence** <*value*>
**match protocol bridge**
**match protocol bridge netbeui**

## Syntax Description

| | |
|---|---|
| **all** | Optional. Specifies matching all UDP port numbers in the specified range (even and odd). Valid only for ip rtp matches. |
| **dscp** <*value*> | Matches IP packets with the specified Differentiated Service Code Point (DSCP) value. Range is 0 to 63. |
| **ip rtp** <*begin port range*> <*end port range*> | Matches RTP packets with even UDP destination port numbers in the specified range (between start and end). |
| **list** <*name*> | Specifies the name of the access-list (ACL) you wish to use to match packets for this QoS map. Refer to *ip access-list extended <name>* on page 492 for more information on creating access-lists. |
| **precedence** <*value*> | Matches IP packets with the specified IP precedence value. Range is 0 through 7. |
| **protocol bridge** | Matches frames being bridged by the router. |
| **protocol bridge netbeui** | Matches only NetBEUI frames being bridged by the router. |

## Default Values

No default value is necessary for this command.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Functional Notes

QoS policies are configured in the AOS CLI to dictate the priority for servicing specified traffic types on a particular interface. QoS policies contain at least one match reference (using the **match** command) and one or more action items (using the **priority**, **bandwidth**, or **set** commands).

The **match** command specifies the criteria used when determining whether incoming traffic is a candidate for the QoS policy action items. Multiple **match** statements can exist within the same QoS policy, allowing a single QoS policy to service various types of traffic.


*Each listed **match** statement is handled independently by the processor. Entering too many **match** statements in a QoS policy can burden the processor.*

For example, consider a network that contains Class A and Class B traffic that each require 25 percent of the total allocated interface bandwidth.

(config)**#qos map MyMap 1**
(config-qos-map)**#match list Class_A**
(config-qos-map)**#match list Class_B**
(config-qos-map)**#bandwidth percent 25**

Alternately, the following configuration is also valid:

(config)**#qos map MyMap 1**
(config-qos-map)**#match list Class_A**
(config-qos-map)**#bandwidth percent 25**

(config)**#qos map MyMap 2**
(config-qos-map)**#match list Class_B**
(config-qos-map)**#bandwidth percent 25**

To remove a configured **match** statement, enter the entire **match** statement with a preceding **no**. For example, to remove the **match** statements from the above configured QoS map:

(config)**#qos map MyMap 1**
(config-qos-map)**#no match list Class_A**

and

(config)**#qos map MyMap 2**
(config-qos-map)**#no match list Class_B**

## Usage Examples

The following example configures QoS for a network with the following needs:

Reserve 15 percent of the line rate for routing traffic and L2 protocol traffic (**max-reserved-bandwidth** = 85)

Line Rate = 512 kbps
Guaranteed 256 kbps for Voice
Guaranteed 96 kbps for Class 1
Guaranteed 52 kbps for Class 2

To configure this QoS policy, enter the following QoS map and interface commands:

1. Allocate LLQ Priority voice traffic

(config)**#qos map MyMap 1**
(config-qos-map)**#match list VOICE**
(config-qos-map)**#priority 256**

2. Allocate the CBWFQ data traffic bandwidth for classes 1 and 2

(config)**#qos map MyMap 2**
(config-qos-map)**#match list CLASS_1**
(config-qos-map)**#bandwidth 96**

(config)**#qos map MyMap 3**
(config-qos-map)**#match list CLASS_2**
(config-qos-map)**#bandwidth 52**

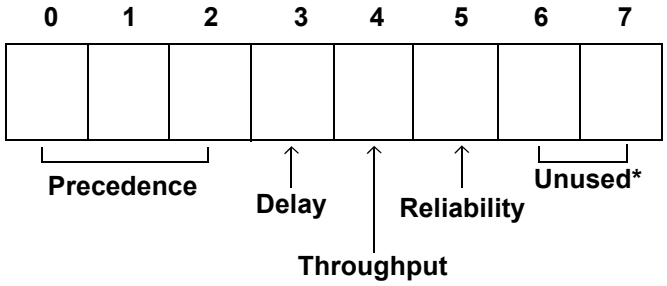
3. Specify the reserved bandwidth and apply the map

(config-fr 1)**#max-reserved-bandwidth 85**
(config-fr 1)**#qos-policy out MyMap**

## Technology Review

RFC791 created a single octet (labeled Type of Service) to help with the difficulty of trying to provide QoS handling in IP networks.

According to RFC791, the Type of Service field contains the following bits:




The three-bit IP precedence values (0 through 7) are specified as:

| | |
|---|---|
| 111 | Network Control Packets |
| 110 | Internetwork Control Packets |
| 101 | Critical Traffic |
| 100 | Flash Override |
| 011 | Flash |
| 010 | Immediate Servicing |
| 001 | Priority Traffic |
| 000 | Routine Data |

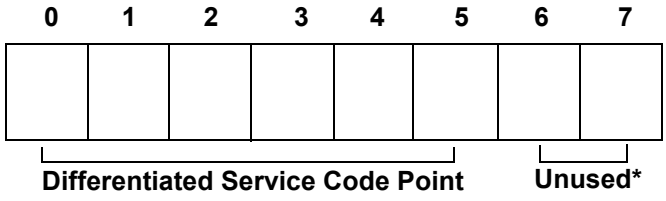
  

The IP Precedence values provide network routers with information about what kind of traffic is contained in the IP packet. Based on the IP Precedence values, some networks (when supported) can offer special handling to certain packets. In addition, providing IP Precedence values to critical traffic (such as route information) ensures that critical packets will always be delivered regardless of network congestion. This traffic is often critical to network and internetwork operation. In general, the higher the IP Precedence value, the more important the traffic and the better handling it should receive in the network. It is important to remember that not all equipment in the public IP network will be configured to recognize and handle IP precedence values. While it is a good idea to set the values for critical traffic, it does not guarantee special handling.

In addition to the precedence values, RFC791 specifies bits for delay, throughput, and reliability to help balance the needs of particular traffic types when traveling on the IP network infrastructure. When these bits are set to 0, they are handled with normal operation. When set to 1, each bit specifies premium handling for that parameter. For example, a 1 in the delay position indicates that the traffic is delay sensitive and care should be taken to minimize delay. A 1 in the throughput position indicates that the traffic has higher bandwidth requirements that should be met. A 1 in the reliability position indicates that the traffic is sensitive to delivery issues and care should be taken to ensure proper delivery with all packets of this type. These extra bits are rarely used because they are quite difficult to balance the cost and benefits of each parameter (especially when more than one bit is set to 1).

The Differentiated Services (DiffServ or DS) model was created in RFC2474 and 2475 to build on the original Type of Service field by creating a six-bit sequence (combining the precedence value with the delay, throughput, and reliability bits). This six-bit sequence increased the number of available values from 8 to 64. The DiffServ model introduced a new concept to QoS in the IP network environment: per-hop behaviors (PHBs). The PHB premise is that equipment using the DiffServ model have an agreed upon set of rules (PHB types) for handling certain network traffic. Though the RFC explicitly defines what each PHB should be capable of, it does not restrict vendor-specific implementation of the PHBs. Each vendor is free to decide how their network product implements the various defined PHBs.

According to RFC2474, the DS field contains the following bits:




*The previously unused bits in the DS field are now used for congestion control and are not discussed int this document.

Equipment following the DiffServ model (DS-compliant nodes) must use the entire six-bit DSCP value to determine the appropriate PHB. The PHBs are defined as the following:

- Default PHB
- Class Selector PHB
- Assured Forwarding PHB (RFC2597)
- Expedited Forwarding PHB (RFC2598)

### Default PHB

All DS-compliant nodes must provide a Default PHB to offer best-effort forwarding service. For Default PHBs, the DSCP value is 0. Any packet that does not contain a standardized DSCP should be mapped to the Default PHB and handled accordingly.

### Class Selector PHB

In the Class Selector PHB, the first three bits in the DSCP value are used for backwards compatibility to systems implementing IP precedence. In this scenario, all but the first three bits of the DS field are set to 0. This compatibility requires DS-compliant nodes to provide the same data services as are provided by nodes implementing IP precedence. The following table is a comparison of IP precedence values to their corresponding DSCP values.

| IP Precedence Value (bits) | DSCP Value (bits) |
|---|---|
| 0 (000) | 0 (000000) |
| 1 (001) | 8 (001000) |
| 2 (010) | 16 (010000) |
| 3 (011) | 24 (011000) |
| 4 (100) | 32 (100000) |
| 5 (101) | 40 (101000) |
| 6 (110) | 48 (110000) |
| 7 (111) | 56 (111000) |

### Assured Forwarding PHB

The flexibility of DiffServ allows for more developed sub-classes of service within each main class using the last three bits of the DSCP. As defined in RFC2597, the Assured Forwarding PHB creates four main classes of service:

| Class | DSCP Bits |
|---|---|
| AF1 | 001XX0 |
| AF2 | 010XX0 |
| AF3 | 011XX0 |
| AF4 | 100XX0 |
| X indicates a do not care value. | |

The first three bits of the DSCP specify the class and the last bit is always zero. Each class is separated into subclasses using the two remaining bits in the DSCP (bits 3 and 4). The subclasses are divided based on the likelihood that packets in the class are dropped in the event of network congestion. The higher the value for bits 3 and 4, the greater the likelihood that the packets will be dropped.

| Bit 3 | Bit 4 | Drop Precedence |
|---|---|---|
| 0 | 1 | Low |
| 1 | 0 | Medium |
| 1 | 1 | High |

The following table lists the Assured Forwarding PHB subclasses and their corresponding DSCP bits and values.

| Class | Subclass | DSCP Bits | DSCP Value |
|---|---|---|---|
| AF1 | 1 | 001010 | 10 |
| | 2 | 001100 | 12 |
| | 3 | 001110 | 14 |
| AF2 | 1 | 010010 | 18 |
| | 2 | 010100 | 20 |
| | 3 | 010110 | 22 |
| AF3 | 1 | 011010 | 26 |
| | 2 | 011100 | 28 |
| | 3 | 011110 | 30 |
| AF4 | 1 | 100010 | 34 |
| | 2 | 100100 | 36 |
| | 3 | 100110 | 38 |

**Expedited Forwarding PHB**

RFC2598 created a new DiffServ PHB intended to provide the best service possible on an IP network. Packets using the Expedited Forwarding PHB markings should be provided service to reduce latency, jitter, dropped packets, and be guaranteed bandwidth during the entire end-to-end transmission journey through the network. The DSCP value for the Expedited Forwarding PHB is 46 (DSCP bits are 101110).

# priority

Use the **priority** command to specify a high-priority queue, prioritizing this traffic above all others. If no traffic is present in any other queue, priority traffic is allowed to burst up to the interface rate; otherwise, priority traffic above the specified bandwidth is dropped. Use the **no** form of this command to disable this feature. Variations of this command include:

**priority** <*bandwidth*>
**priority** <*bandwidth*> <*burst*>
**priority percent** <*bandwidth*>
**priority unlimited**

## Syntax Description

| | |
|---|---|
| <*bandwidth*> | Specifies a low latency queue, prioritizing this traffic above all other user traffic in kilobits per second. Range is 8 to 1,000,000. If no traffic is present in any other queue, priority traffic is allowed to burst up to the interface rate. Otherwise, priority traffic above the specified bandwidth will be dropped. |
| <*burst*> | Optional. Specifies the maximum burst size (in bytes) for traffic in this priority queue. This parameter should be left unconfigured for optimal performance. Range: 3 to 1,000,000. |
| **percent** <*bandwidth*> | Allocates a minimum bandwidth for a traffic class, specifying the minimum as a percentage of the total interface bandwidth. This command is especially useful for protecting bandwidth allocation in multilink applications. See *Functional Notes* for more details. |
| **unlimited** | Optional. Specifies no limits on the priority queue bandwidth. Excessive traffic matching the QoS map can potentially use all of the available bandwidth on theinterface, so use this feature with caution. |

## Default Values

No default value is necessary for this command.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Functional Notes

The priority queue is intended for constant bit rate traffic such as voice due to the rate limiting. The sum of the bandwidths reserved by priority commands for all entries of a QoS map cannot exceed the **max-reserved-bandwidth** rate specified for the interfaces that the map is applied to. Priority bandwidth is guaranteed bandwidth (in kbps).

**Determining Bandwidth Entries**

*When possible, use the **priority** <bandwidth> command to specify an absolute amount of bandwidth (in kbps) for the priority queue.*

When determining the **priority percent** <*bandwidth*> entry, use the following formula:

$$\frac{\text{Bandwidth}}{\text{Line Rate}}$$

where

| | |
|---|---|
| *Bandwidth* | Specifies the minimum amount of bandwidth needed for the traffic (in kbps). |
| *Line Rate* | Specifies the total data rate configured on the interface (for example, 8 DS0s (64 kbps per DS0) on a T1 equals a line rate of 512 kbps). |

For example, to specify 80 kbps of data on an interface with a total of 512 kbps of available bandwidth, enter the following command:

(config-qos-map 1)**#priority percent 16**

## Usage Examples

The following example configures QoS for a network with the following needs:

Reserve 15 percent of the line rate for routing traffic and L2 protocol traffic (**max-reserved-bandwidth** = 85)

Line Rate = 512 kbps
Guaranteed 256 kbps for Voice
Guaranteed 96 kbps for Class 1
Guaranteed 52 kbps for Class 2

To configure this QoS policy, enter the following QoS map and interface commands:

1. Allocate LLQ Priority voice traffic

(config)**#qos map MyMap 1**
(config-qos-map)**#match list VOICE**
(config-qos-map)**#priority 256**

2. Allocate the CBWFQ data traffic bandwidth for classes 1 and 2

(config)**#qos map MyMap 2**
(config-qos-map)**#match list CLASS_1**
(config-qos-map)**#bandwidth 96**
(config)**#qos map MyMap 3**
(config-qos-map)**#match list CLASS_2**
(config-qos-map)**#bandwidth 52**

3. Specify the reserved bandwidth and apply the map

(config-fr 1)**#max-reserved-bandwidth 85**
(config-fr 1)**#qos-policy out MyMap**

# set dscp <*value*>

Use the **set dscp** command to modify the Differentiated Service Code Point (DSCP) field (on matching packets) to the specified value. For more details on determining the DSCP field, refer to the *Technology Review* section of the *match* on page 1988. Use the **no** form of this command to remove a specified DSCP value.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the decimal DSCP value. Range is 0 to 63. |

## Default Values

No default value is necessary for this command.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Functional Notes

QoS policies are configured in the AOS CLI to dictate the priority for servicing specified traffic types on a particular interface. QoS policies contain at least one match reference (using the **match** command) and one or more action items (using the **priority**, **bandwidth**, or **set** commands).

The **set dscp** command can be used to change the Differentiated Services (DS) Field for incoming traffic serviced by the QoS policy. Every IPv4 header contains an 8-bit Type of Service (ToS) field used for marking data types requiring special handling when traveling through the network. Originally this ToS field was used for IP precedence markings (using only the first three bits of the eight-bit field), and was later revised in RFC2474 to create the six-bit DS field (reserving the last two bits of the field for future use). The DS field can be manipulated to indicate higher or lower traffic priority using decimal values between 0 and 63.

## Usage Examples

This command sets the DSCP value (for all matching traffic) to **46**:

(config)**#qos map VOICEMAP 10**
(config-qos-map)**#set dscp 46**

# set precedence <*value*>

Use the **set precedence** command to modify the IP Precedence value (on matching packets) to the specified value. For more details on determining the IP Precedence value, refer to the *Technology Review* section of the *match* on page 1988. Use the **no** form of this command to remove a specified precedence.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the decimal IP precedence value. Range is 0 to 7. |

## Default Values

No default value is necessary for this command.

## Command History

| | |
|---|---|
| Release 6.1 | Command was introduced. |

## Functional Notes

QoS policies are configured in the AOS CLI to dictate the priority for servicing specified traffic types on a particular interface. QoS policies contain at least one match reference (using the **match** command) and one or more action items (using the **priority**, **bandwidth**, or **set** commands).

The **set dscp** command can be used to change the Differentiated Services (DS) Field for incoming traffic serviced by the QoS policy. Every IPv4 header contains an 8-bit Type of Service (ToS) field used for marking data types requiring special handling when traveling through the network. Originally this ToS field was used for IP precedence markings (using only the first three bits of the eight-bit field), and was later revised in RFC2474 to create the six-bit DS field (reserving the last two bits of the field for future use). The DS field can be manipulated to indicate higher or lower traffic priority using decimal values between 0 and 63.

## Usage Examples

This command sets the IP precedence value (for all matching traffic) to **5**:

(config)**#qos map VOICEMAP 10**
(config-qos-map)**#set precedence 5**

# RADIUS GROUP COMMAND SET

To activate the Radius Group Configuration mode, enter the **aaa group server** command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#aaa group server radius myServer
(config-sg-radius)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:

*cross-connect* *on page 29*
*description <text>* *on page 32*
*do* *on page 33*
*end* *on page 34*
*exit* *on page 35*
*shutdown* *on page 36*

All other commands for this command set are described in this section in alphabetical order.

*server* *on page 2000*

# server

Use the **server** command to add a predefined RADIUS server to the current named list of servers. Use the **no** form of this command to disable this feature. Refer to *radius-server* on page 646 for more information. Variations of this command include:

**server** <*number*>
**server acct-port** <*number*>
**server auth-port** <*number*>

## Syntax Description

| | |
|---|---|
| **acct-port** <*number*> | Optional. Specifies the accounting port (by number) to add to the list. |
| **auth-port** <*number*> | Optional. Specifies the authorization port (by number) to add to the list. |

## Default Values

No defaults necessary for this command.

## Command History

| | |
|---|---|
| Release 5.1 | Command was introduced. |

## Usage Examples

The following example adds a server to the **myServers** list:

(config)**#aaa group server radius myServers**
(config-sg-radius)**#server 1.2.3.4 acct-port 786 auth-port 1812**
(config-sg-radius)**#server 4.3.2.1**
(config-sg-radius)**#exit**
(config)#

or

(config)**#aaa group server radius myServers**
(config-sg-radius)**#server 4.3.2.1**
(config-sg-radius)**#exit**
(config)#

# ROUTE MAP CONFIGURATION COMMAND SET

To activate the Route Map Configuration mode, enter the **route-map** command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#route-map MyMap permit 100
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:



All other commands for this command set are described in this section in alphabetical order.



# match as-path <*name*>

Use the **match as-path** command to configure the route map to route traffic based on the AS path list name. Use the **no** form of this command to discontinue matching.

## Syntax Description

| | |
|---|---|
| <*name*> | Specifies the name of the AS path list you want to match. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example instructs the route map named **MyMap** to match the AS path list named **TestPath**:

(config)**#route-map MyMap permit 100**
(config-route-map)**#match as-path TestPath**

# match community <*name*>

Use the **match community** command to configure the route map to route traffic based on a specified community. Use the **no** form of this command to discontinue matching. Variations of this command include:

**match community** <*name*>
**match community** <*name*> **exact-match**

## Syntax Description

| | |
|---|---|
| <*name*> | Specifies the name of the community you want to match. |
| **exact-match** | Optional. Specifies that the route map must match the community name exactly. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example instructs the route map named **MyMap** to match the community named **MyCommunity**:

(config)**#route-map MyMap permit 100**
(config-route-map)**#match community MyCommunity**

# match ip address <*name*>

Use the **match ip address** command to configure the route map to route traffic based on the access control list name defined with the **ip access-list** command. Refer to *ip access-list extended <name>* on page 492 for more information. Use the **no** form of this command to discontinue matching.

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies the name of the access control list to match. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example instructs the route map named **MyMap** to match the IP address access control list named **MyList**:

(config)**#route-map MyMap permit 100**
(config-route-map)**#match ip address MyList**

# match ip address prefix-list *<name>*

Use the **match ip address prefix-list** command to configure the route map to route traffic based on a prefix list route filter. The name of the prefix list is defined with the **ip prefix-list** command. Refer to *ip prefix-list <name> description <"text">* on page 560 for more information. Use the **no** form of this command to discontinue matching.

## Syntax Description

| | |
|---|---|
| *<name>* | Specifies matching the IP address based on the prefix list name. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example instructs the route map named **MyMap** to match the IP address prefix list named **MyList**:

(config)**#route-map MyMap permit 100**
(config-route-map)**#match ip address prefix-list MyList**

# match ip dscp

Use the **match ip dscp** command to configure the route map to route traffic based on the Differentiated Services Code Point (DSCP) value in the IP header of the packet. Use the **no** form of this command to discontinue matching. Variations of this command include:

**match ip dscp** *<value>*
**match ip dscp afxx**
**match ip dscp csx**
**match ip dscp default**
**match ip dscp ef**

## Syntax Description

| | |
|---|---|
| *<value>* | Specifies the DSCP numeric value to match. Valid range is 0 to 63. |
| **afxx** | Specifies the assured forwarding (AF) value to match. Select from 11, 12, 13, 21, 22, 23, 31, 32, 33, 41, 42, or 43. |
| **csx** | Specifies the class selector (CS) value to match. Valid range is 1 to 7. |
| **default** | Specifies matching the default IP DSCP value. |
| **ef** | Specifies matching those packets marked for expedited forwarding (EF). |

## Default Values

No default value necessary for this command.

## Command History

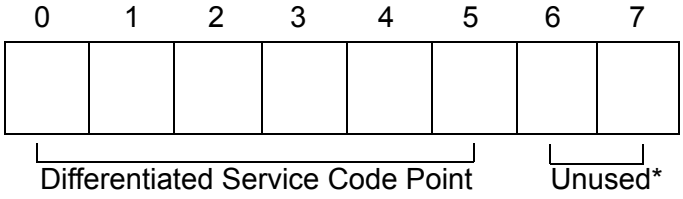
| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Functional Notes

The Differentiated Services (DiffServ or DS) model was created in RFC2474 and 2475 to build on the original Type of Service field by creating a six-bit sequence (combining the precedence value with the delay, throughput, and reliability bits). This six-bit sequence increased the number of available values from 8 to 64. The DiffServ model introduced a new concept to QoS in the IP network environment: per-hop behaviors (PHBs). The PHB premise is that pieces equipment using the DiffServ model have an agreed upon set of rules (PHB types) for handling certain network traffic. Though the RFC explicitly defines what each PHB should be capable of, it does not restrict vendor-specific implementation of the PHBs. Each vendor is free to decide how their network product implements the various defined PHBs.

According to RFC2474, the DS field contains the following bits:




* The previously unused bits in the DS field are now used for congestion control and are not discussed in this document.

Equipment following the DiffServ model (DS-compliant nodes) must use the entire six-bit DSCP value to determine the appropriate PHB. The PHBs are defined as the following:

- Default PHB
- Class Selector PHB
- Assured Forwarding PHB (RFC2597)
- Expedited Forwarding PHB (RFC2598)

**Default PHB**

All DS-compliant nodes must provide a Default PHB to offer best-effort forwarding service. For Default PHBs, the DSCP value is 0. Any packet that does not contain a standardized DSCP should be mapped to the Default PHB and handled accordingly.

**Class Selector PHB**

In the Class Selector PHB, the first three bits in the DSCP value are used for backwards compatibility to systems implementing IP precedence. In this scenario, all but the first three bits of the DS field are set to 0. This compatibility requires DS-compliant nodes to provide the same data services as are provided by nodes implementing IP precedence. The following table is a comparison of IP precedence values to their corresponding DSCP values.

| IP Precedence Value (bits) | DSCP Value (bits) |
|---|---|
| 0 (000) | 0 (000000) |
| 1 (001) | 8 (001000) |
| 2 (010) | 16 (010000) |
| 3 (011) | 24 (011000) |
| 4 (100) | 32 (100000) |
| 5 (101) | 40 (101000) |
| 6 (110) | 48 (110000) |
| 7 (111) | 56 (111000) |

### Assured Forwarding PHB

The flexibility of DiffServ allows for more developed subclasses of service within each main class using the last three bits of the DSCP. As defined in RFC2597, the Assured Forwarding PHB creates four main classes of service:

| Class | DSCP Bits |
|---|---|
| AF1 | 001XX0 |
| AF2 | 010XX0 |
| AF3 | 011XX0 |
| AF4 | 100XX0 |
| X indicates a “do not care” value | |

The first three bits of the DSCP specify the class, and the last bit is always zero. Each class is separated into subclasses using the two remaining bits in the DSCP (bits 3 and 4). The subclasses are divided based on the likelihood that packets in the class are dropped in the event of network congestion. The higher the value for bits 3 and 4, the greater the likelihood that the packets will be dropped.

| Bit 3 | Bit 4 | Drop Precedence |
|---|---|---|
| 0 | 1 | Low |
| 1 | 0 | Medium |
| 1 | 1 | High |

The following table lists the Assured Forwarding PHB subclasses and their corresponding DSCP bits and values.

| Class | Subclass | DSCP Bits | DSCP Value |
|---|---|---|---|
| AF1 | 1 | 001010 | 10 |
| | 2 | 001100 | 12 |
| | 3 | 001110 | 14 |
| AF2 | 1 | 010010 | 18 |
| | 2 | 010100 | 20 |
| | 3 | 010110 | 22 |
| AF3 | 1 | 011010 | 26 |
| | 2 | 011100 | 28 |
| | 3 | 011110 | 30 |
| AF4 | 1 | 100010 | 34 |
| | 2 | 100100 | 36 |
| | 3 | 100110 | 38 |

**Expedited Forwarding PHB**

RFC2598 created a new DiffServ PHB intended to provide the best service possible on an IP network. Packets using the Expedited Forwarding PHB markings should be provided service to reduce latency, jitter, and dropped packets, and should be guaranteed bandwidth during the entire end-to-end transmission journey through the network. The DSCP value for the Expedited Forwarding PHB is 46 (DSCP bits are 101110).

## Usage Examples

The following example instructs the route map named **MyMap** to match the IP header with a DSCP Assured Forwarding Class 1, Subclass 2 (**af12**):

(config)**#route-map MyMap permit 100**
(config-route-map)**#match ip dscp af12**

# match ip precedence

Use the **match ip precedence** command to configure the route map to route traffic based on the precedence value in the IP header of the packet. Use the **no** form of this command to discontinue matching. Variations of this command include:

**match ip precedence** <*value*>
**match ip precedence critical**
**match ip precedence flash**
**match ip precedence flash-override**
**match ip precedence immediate**
**match ip precedence internet**
**match ip precedence network**
**match ip precedence priority**
**match ip precedence routine**

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies matching the IP precedence (in numeric value). Valid range is 0 to 7 in ascending order of importance. |
| **routine** | Specifies matching the IP precedence **routine**. (Numeric value of 0.) |
| **priority** | Specifies matching the IP precedence **priority**. (Numeric value of 1.) |
| **immediate** | Specifies matching the IP precedence **immediate**. (Numeric value of 2.) |
| **flash** | Specifies matching the IP precedence **flash**. (Numeric value of 3.) |
| **flash-override** | Specifies matching the IP precedence **flash-override**. (Numeric value of 4.) |
| **critical** | Specifies matching the IP precedence **critical**. (Numeric value of 5.) |
| **internet** | Specifies matching the IP precedence **internet**. (Numeric value of 6.) This level is reserved for internal network use. |
| **network** | Specifies matching the IP precedence **network**. (Numeric value of 7.) This level is reserved for internal network use. |

## Default Values

No default value necessary for this command.

## Command History

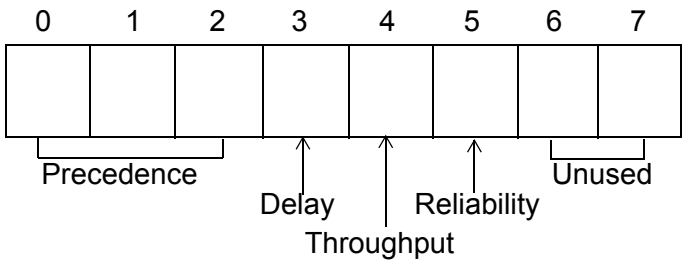
| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Functional Notes

RFC791 created a single octet (labeled Type of Service) to help with the difficulty of trying to provide QoS handling in IP networks.

According to RFC791, the Type of Service field contains the following bits:




The three-bit IP precedence values (0 through 7) are specified as:

| | |
|---|---|
| 111 | Network Control Packets |
| 110 | Internetwork Control Packets |
| 101 | Critical Traffic |
| 100 | Flash Override |
| 011 | Flash |
| 010 | Immediate Servicing |
| 001 | Priority Traffic |
| 000 | Routine Data |

The IP precedence values provide network routers with information about the kind of traffic contained in the IP packet. Based on the IP precedence values, some networks (when supported) can offer special handling to certain packets. In addition, providing IP precedence values to critical traffic (such as route information) ensures that critical packets will always be delivered regardless of network congestion. This traffic is often critical to network and internetwork operation. In general, the higher the IP precedence value, the more important the traffic and the better handling it should receive in the network. It is important to remember that not all equipment in the public IP network will be configured to recognize and handle IP precedence values. While it is a good idea to set the values for critical traffic, it does not guarantee special handling.

In addition to the IP precedence values, RFC791 specifies bits for delay, throughput, and reliability to help balance the needs of particular traffic types when traveling on the IP network infrastructure. When these bits are set to 0, they are handled with normal operation. When set to 1, each bit specifies premium handling for that parameter. For example, a 1 in the delay position indicates that the traffic is delay-sensitive and care should be taken to minimize delay. A 1 in the throughput position indicates that the traffic has higher bandwidth requirements that should be met. A 1 in the reliability position indicates that the traffic is sensitive to delivery issues and care should be taken to ensure proper delivery with all packets of this type. These extra bits are rarely used because it is quite difficult to balance the cost and benefits of each parameter (especially when more than one bit is set to 1).

## Usage Examples

The following example instructs the route map named **MyMap** to match the IP precedence value of **critical**:

(config)**#route-map MyMap permit 100**
(config-route-map)**#match ip precedence critical**

# match length *<minimum> <maximum>*

Use the **match length** command to configure the route map to route traffic based on the packet length. Use the **no** form of this command to discontinue matching.

## Syntax Description

| | |
|---|---|
| *<minimum>* | Specifies the minimum packet length you want to match. Valid range is 1 to 4,294,967,295.) |
| *<maximum>* | Specifies the maximum packet length you want to match. Valid range is 1 to 4,294,967,295.) |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example instructs the route map named **MyMap** to match packets with a minimum length of **1** and a maximum length of **200**:

(config)**#route-map MyMap permit 100**
(config-route-map)**#match length 1 200**

# match metric <*value*>

Use the **match metric** command to configure the route map to route traffic based on a specified metric value. Use the **no** form of this command to discontinue matching.

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the metric value you want to match. Valid range is 1 to 4,294,967,295. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example instructs the route map named **MyMap** to match the metric value **100**:

(config)**#route-map MyMap permit 100**
(config-route-map)**#match metric 100**

# set as-path prepend

Use the **set as-path prepend** command to prepend a number to the AS path to influence the best-path selection process by making the AS path appear further away. Use the **no** form of this command to disable this feature. Variations of this command include:

**set as-path prepend** <*number*>
**set as-path prepend last-as** <*number*>

## Syntax Description

| | |
|---|---|
| **prepend** <*number*> | Specifies a number to be prepended to the AS path value as an autonomous number. Valid range is 1 to 65,535. |
| **prepend last-as** <*number*> | Specifies a number to be prepended to the last AS path number. Valid range is 1 to 10. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example prepends the number **2** to the last AS path number:

(config)**#route-map MyMap permit 100**
(config-route-map)**#set as-path prepend last-as 2**

# set comm-list <*name*> delete

Use the **set comm-list delete** command to specify a list of communities to delete. Use the **no** form of this command to disable this feature.

## Syntax Description

| | |
|---|---|
| <*name*> | Specifies the name of the list of communities to delete. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example deletes the community list named **listname**:

(config)**#route-map MyMap permit 100**
(config-route-map)**#set comm-list listname delete**

# set community

Use the **set community** command to modify the community attribute for all paths serviced by the route map. Use the **no** form of this command to disable this feature. Variations of this command include:

**set community** <*value*>
**set community** <*value*> **add**
**set community** <*value*> **internet**
**set community** <*value*> **local-as**
**set community** <*value*> **no-advertise**
**set community** <*value*> **no-export**
**set community none**

## Syntax Description

| | |
|---|---|
| <*value*> | Sets the community attribute to the specified community number for routes serviced by this route map. This is a numeric value that can be an integer from 1 to 4,294,967,295 or string in the form “aa:nn”, where the value of “aa” is the AS number and the value of “nn” is the community number. Multiple community-number parameters can be present in the command. |
| **add** | Appends the listed community number to the end of the community attribute for routes serviced by this route map. |
| **internet** | Sets the community attribute to the INTERNET community number for routes serviced by this route map. |
| **local-as** | Sets the community attribute to the NO_EXPORT_SUBCONFED community number for routes serviced by this route map. Routes containing this attribute should not be advertised to external BGP peers. |
| **no-advertise** | Sets the community attribute to the NO_ADVERTISE community number for routes serviced by this route map. Routes containing this attribute should not be advertised to any BGP peer. |
| **no-export** | Sets the community attribute to the NO_EXPORT community number for routes serviced by this route map. Routes containing this attribute should not be advertised to BGP peers outside a confederation boundary. |
| **none** | Removes all communities from BGP routes serviced by this route map. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example sets the community number for BGP routes to the Internet community:

(config)**#route-map MyMap permit 100**
(config-route-map)**#set community internet**

# set default interface

Use the **set default interface** command to specify a default interface to redirect traffic to the specified interface if there is no specific routing information for the traffic. If more than one interface is specified, the router uses the first available interface from the list. Use the **no** form of this command to remove the default interface. Variations of this command include:

**set default interface** <*interface*>
**set default interface null 0**

## Syntax Description

| | |
|---|---|
| <*interface*> | Specifies the default interface. Specify an interface in the format <*interface type [slot/port | slot/port.sub-interface id | interface id | interface id.sub-interface id]*>. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**.Type **set default interface ?** for a list of valid interface types. |
| **null 0** | Redirects traffic to the specified interface regardless of available routing information. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example sets the default interface as **ppp 1** interface:

(config)**#route-map MyMap permit 100**
(config-route-map)**#set default interface ppp 1**

# set interface <*interface*>

Use the **set interface** command to specify an output interface for the packet. Multiple interfaces can be specified. The router forwards the packet along the first usable interface. Use the **no** form of this command to cancel output from the specified interface.

## Syntax Description

| | |
|---|---|
| <*interface*> | Sets output interface type for the packet. Specify an interface in the format <*interface type [slot/port | slot/port.sub-interface id | interface id | interface id.sub-interface id]*>. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **set interface ?** for a list of valid interfaces. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example sets the output interface as PPP 1:

(config)**#route-map MyMap permit 100**
(config-route-map)**#set interface ppp 1**

# set ip default next-hop *<interface>*

Use the **set ip default next-hop** command to set the next-hop IP address to the specified interface's address for all routes serviced by the route map that do no have explicit routing information available. Use the **no** form of this command to remove the configured default next-hop.

## Syntax Description

| | |
|---|---|
| *<interface>* | Specifies the default interface. Specify an interface in the format *<interface type [slot/port \| slot/port.sub-interface id \| interface id \| interface id.sub-interface id]>*. For example, for a T1 interface use **t1 0/1**; for an Ethernet sub-interface, use **eth 0/1.1**; for a PPP interface, use **ppp 1**; and for an ATM sub-interface use **atm 1.1**. Type **set default next-hop ?** for a list of valid interface types. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example sets the default next-hop interface to the **ppp 1** interface:

(config)**#route-map MyMap permit 100**
(config-route-map)**#set ip default next-hop ppp 1**

# set ip df

Use the **set ip df** command to identify the packet as “don’t fragment” (DF). Use the **no** form of this command to remove this designation.

## Syntax Description

No subcommands.

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example designates the packet as “don’t fragment”:

(config)**#route-map MyMap permit 100**
(config-route-map)**#set ip df**

# set ip dscp

Use the **set ip dscp** command to configure the route map to set the Differentiated Services Code Point (DSCP) value in the IP header of the packet for traffic serviced by this route map. For more details on DSCP values, refer to the command *match ip dscp* on page 2006. Use the **no** form of this command to remove the specified DSCP value. Variations of this command include:

**set ip dscp** <*value*>
**set ip dscp afxx**
**set ip dscp csx**
**set ip dscp default**
**set ip dscp ef**

## Syntax Description

| | |
|---|---|
| <*value*> | Specifies the DSCP numeric value. Valid range is 0 to 63. |
| **afxx** | Specifies the assured forwarding (AF) class and subclass. Select from: 11 (001010), 12 (001100), 13 (001110), 21 (010010), 22 (010100), 23 (010110), 31 (011010), 32 (011100), 33 (011110), 41 (100010), 42 (100100), or 43 (100110). |
| **csx** | Specifies the class selector (CS) value. Valid range is 1 to 7. |
| **default** | Specifies the default IP DSCP value (000000). |
| **ef** | Specifies marking for expedited forwarding (EF). |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example instructs the route map named **MyMap** to set the IP header with a DSCP Assured Forwarding Class 1, Subclass 2 (**af12**):

(config)**#route-map MyMap permit 100**
(config-route-map)**#set ip dscp af12**

# set ip next-hop <*ip address*>

Use the **set ip next-hop** command to set the next-hop IP address to the specified address for all routes serviced by the route map. Use the **no** form of this command to remove the configured next-hop address.

## Syntax Description

| | |
|---|---|
| <*ip address*> | Specifies a valid IP address. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). More than one address can be entered, and the router uses the first available route from the list. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example sets the ip next-hop interface to **10.10.11.254** in the header of the route map named **MyMap**:

(config)**#route-map MyMap permit 100**
(config-route-map)**#set ip next-hop 10.10.11.254**

# set ip precedence

Use the **set ip precedence** command to configure the route map to set the precedence value in the IP header of the packet for traffic serviced by the route map. For more details on IP Precedence values, see the command *match ip precedence* on page 2010. Use the **no** form of this command to removed the specified IP Precedence value. Variations of this command include:

**set ip precedence** <*value*>
**set ip precedence critical**
**set ip precedence flash**
**set ip precedence flash-override**
**set ip precedence immediate**
**set ip precedence internet**
**set ip precedence network**
**set ip precedence priority**
**set ip precedence routine**

## Syntax Description

| | |
|---|---|
| <*value*> | Sets the IP precedence (in numeric value). Valid range is 0 to 7 in ascending order of importance. |
| **critical** | Sets the IP precedence as **critical**. (Numeric value of 5.) |
| **flash** | Sets the IP precedence as **flash**. (Numeric value of 3.) |
| **flash-override** | Sets the IP precedence as **flash-override**. (Numeric value of 4.) |
| **immediate** | Sets the IP precedence as **immediate**. (Numeric value of 2.) |
| **internet** | Sets the IP precedence as **internet**. (Numeric value of 6.) This level is reserved for internal network use. |
| **network** | Sets the IP precedence as **network**. (Numeric value of 7.) This level is reserved for internal network use. |
| **priority** | Sets the IP precedence as **priority**. (Numeric value of 1.) |
| **routine** | Sets the IP precedence as **routine**. (Numeric value of 0.) |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

## Usage Examples

The following example sets an IP precedence value of **critical** in the IP header of the route map named **MyMap**:

(config)**#route-map MyMap permit 100**
(config-route-map)**#set ip precedence critical**

# set local-preference *<value>*

Use the **set local-preference** command to restrict traffic to a local autonomous system. Use the **no** form of this command to cancel the local preference.

## Syntax Description

| | |
|---|---|
| *<value>* | Sets the local preference value. Valid range is 0 to 4,294,967,295. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example sets the local preference for **MyMap** to a value of **100**:

(config)**#route-map MyMap permit 100**
(config-route-map)**#set local-preference 100**

# set metric *<value>*

Use the **set metric** command to specify a metric value for the route map. Use the **no** form of this command to cancel the metric value.

## Syntax Description

| | |
|---|---|
| *<value>* | Sets the metric value. Valid range is 0 to 4,294,967,295. |

## Default Values

No default value necessary for this command.

## Command History

| | |
|---|---|
| Release 10.1 | Command was introduced. |

## Usage Examples

The following example sets the metric value for **MyMap** to **100**:

(config)**#route-map MyMap permit 100**
(config-route-map)**#set metric 100**

# TACACS+ GROUP CONFIGURATION COMMAND SET

To activate the Terminal Access Controller Access Control System Plus (TACACS+) Group Configuration mode, enter the **aaa group server tacacs+** command at the Global Configuration mode prompt. For example:

```
>enable
#configure terminal
(config)#aaa group server tacacs+ TEST GROUP
(config-sg-tacacs+)#
```

The following commands are common to multiple command sets and are covered in a centralized section of this guide. For more information, refer to the sections listed below:

*cross-connect* *on page 29*
*do* *on page 33*
*end* *on page 34*
*exit* *on page 35*

All other commands for this command set are described in this section in alphabetical order.

*server <value>* *on page 2028*

## **server** <*value*>

Use the **server** command to specify a particular TACACS+ server's IP address or host name. Use the **no** form of this command to disable this feature.

### Syntax Description

| | |
|---|---|
| <*value* | Specifies a TACACS+ server IP address or host name. IP addresses should be expressed in dotted decimal notation (for example, **10.10.10.1**). |

### Default Values

No default is necessary for this command.

### Command History

| | |
|---|---|
| Release 11.1 | Command was introduced. |

### Usage Examples

The following example specifies the IP address of the TACACS+ server:

(config)**#aaa group server tacacs+ TEST_GROUP**
(config-sg-tacacs+)**#server 192.168.1.1**
(config-sg-tacacs+)#

# Index



## E

