Configuring RADIUS Authentication for Device Administration

Introduction

The use of AAA services (Authentication, Authorization, and Accounting) allows for several methods of controlling and recording access to AOS-based devices. The two methods of achieving this result involve either RADIUS or TACACS+ servers. This guide will specifically cover the use of controlling login to the administrative interface of an AOS-based device using RADIUS authentication.

Command Line Configuration

Enabling the Service & Defining a Server

AOS-based devices require that the AAA process be enabled and that at least one RADIUS server is defined. The AOS device and the RADIUS server being accessed must agree on the Pre-Shared Key for the process to operate successfully; the key is used to encrypt the password portion of the RADIUS authentication request message. The configuration is as such:

```
aaa on
!
radius-server host <RADIUS Server IP> key <Pre-Shared Key>
```

NOTE: *Working with multiple AAA servers is covered under a different document.*

Define Authentication Methods

The next step is to define the desired authentication methods. This can be done to all interfaces using one command that defines the default list, or by specifying specific lists that can be applied to individual administrative services.

The lists define the order of operations for authentication. The primary method will only fail over to the next defined method if the previous method is unavailable. For instance, if the RADIUS server is specified first and then the Local-User List, the Local-User List will only be consulted if the RADIUS server does not respond to RADIUS request messages. If the RADIUS server rejects the user credentials, the user will be denied access and the Local-User List will not be consulted.

If there are no valid methods left in a given list, then the user will be denied access. An example is if only the RADIUS server is defined within the list, and the RADIUS server cannot be contacted for any reason; user login will not be possible because there are no longer any valid methods within the list. For this reason, it is always recommended to end every list with the Local-User List so the device can still be accessed if communication with the RADIUS server is interrupted because the Local-User List will never be unavailable.

The following example will apply the “default” authentication method as RADIUS authentication first and the Local-User List second in the event RADIUS communication fails. This will affect all administrative services that do not have a defined login method:

```
aaa authentication login default group radius local
```

The following will show the same authentication method used as a named list:

```
aaa authentication login LoginUseRadiusLocal group radius local
```

Apply Authentication Methods to Services

The next step is to apply the defined named authentication method list to the specific interfaces, if the “default” list method is not used.

NOTE: *The HTTP authentication method will apply to both HTTP and HTTPS connections to the router.*

```
line con 0
  login authentication LoginUseRadiusLocal
!
line telnet 0 4
  login authentication LoginUseRadiusLocal
!
line ssh 0 4
  login authentication LoginUseRadiusLocal
```

```
!
ip http authentication LoginUseRadiusLocal
!
ftp authentication LoginUseRadiusLocal
```

Defining the Enable Password Method

The enable password is uniquely associated with the command-line interface, and is irrelevant to the GUI. If command-line access is not being made available for user access or if RADIUS enable authentication is not required, then this section is optional.

Regardless of whether RADIUS or TACAS+ servers are used, they are inherently username & password systems. This poses a specific challenge for enable authentication, where the user is not asked for a username when entering enable mode. Both RADIUS and TACACS+ therefore define a username to be sent with the enable password authentication requests; RADIUS allows for this username being custom-defined.

The authentication methodology is the same as before, and it is recommended in this case that you define the final method as the locally-configured enable password, as such:

```

radius-server enable-username <Enable Username>
!
aaa authentication enable default group radius enable
```

Web Interface Configuration

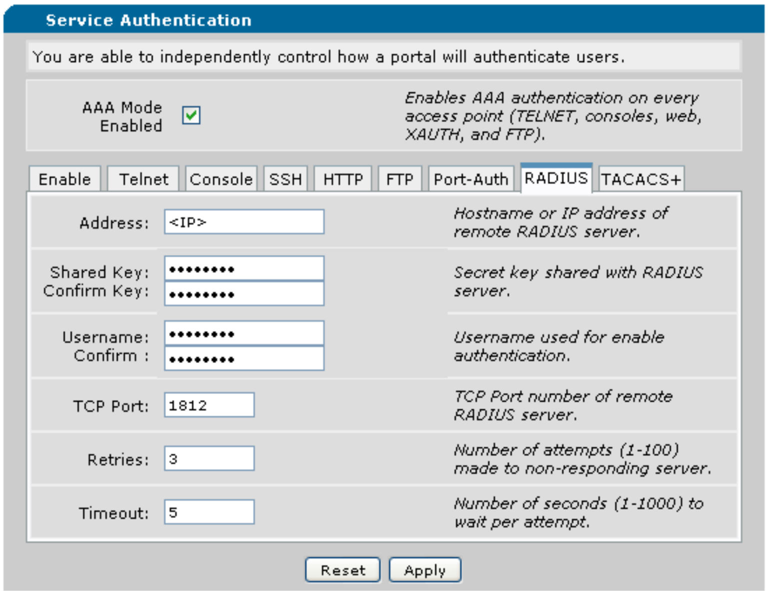
This section will define the methods for configuring this functionality through the GUI. The technology definitions and explanations will not be repeated; please refer to the relevant command line configuration section for more information.

The AAA configuration is accomplished from the “System → Passwords” page, in the bottom section entitled “Service Authentication”.

NOTE: *The functionality allowed by the GUI is limited in that it allows for only one method to be defined, and it uses pre-defined names for its authentication lists. This means that it is not possible to have the Local-User List as a fallback position should the RADIUS server be unavailable. For this reason, CLI configuration is recommended.*

Enabling the Service, Defining a Server, & the Enable Username

The “AAA Mode Enabled” checkbox must be checked and the RADIUS server defined with a Pre-Shared Key & optionally the Enable Username, as shown:



You are able to independently control how a portal will authenticate users.

AAA Mode Enabled ☑ — *Enables AAA authentication on every access point (TELNET, consoles, web, XAUTH, and FTP).*

Address: <IP> — *Hostname or IP address of remote RADIUS server.*

Shared Key: •••••••• Confirm Key: •••••••• — *Secret key shared with RADIUS server.*

Username: •••••••• Confirm : •••••••• — *Username used for enable authentication.*

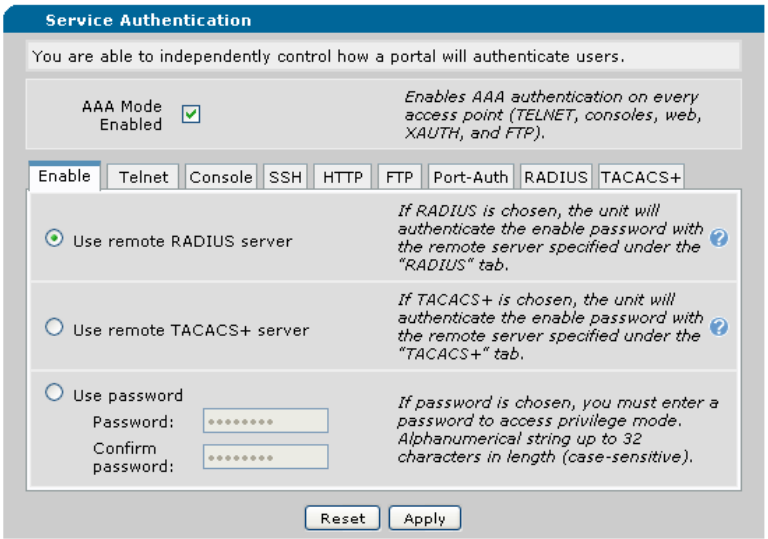
TCP Port: 1812 — *TCP Port number of remote RADIUS server.*

Retries: 3 — *Number of attempts (1-100) made to non-responding server.*

Timeout: 5 — *Number of seconds (1-1000) to wait per attempt.*

Define & Apply Authentication Methods

The GUI does not support the use of a “default” list. Each service will have one of the predefined lists that the GUI creates applied to it, based upon the bullet-point chosen under the Service’s tab. Several examples are shown below:



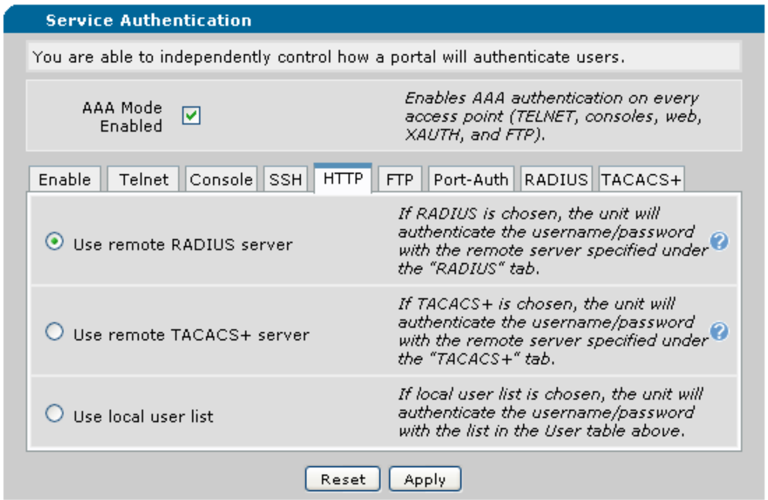
You are able to independently control how a portal will authenticate users.

AAA Mode Enabled ☑ — *Enables AAA authentication on every access point (TELNET, consoles, web, XAUTH, and FTP).*

● Use remote RADIUS server — *If RADIUS is chosen, the unit will authenticate the enable password with the remote server specified under the “RADIUS” tab.*

○ Use remote TACACS+ server — *If TACACS+ is chosen, the unit will authenticate the enable password with the remote server specified under the “TACACS+” tab.*

○ Use password
Password: ••••••••
Confirm password: ••••••••
— *If password is chosen, you must enter a password to access privilege mode. Alphanumerical string up to 32 characters in length (case-sensitive).*



Service Authentication

You are able to independently control how a portal will authenticate users.

AAA Mode Enabled ☑ — *Enables AAA authentication on every access point (TELNET, consoles, web, XAUTH, and FTP).*

● Use remote RADIUS server — *If RADIUS is chosen, the unit will authenticate the username/password with the remote server specified under the "RADIUS" tab.*

○ Use remote TACACS+ server — *If TACACS+ is chosen, the unit will authenticate the username/password with the remote server specified under the "TACACS+" tab.*

○ Use local user list — *If local user list is chosen, the unit will authenticate the username/password with the list in the User table above.*

Configuring the RADIUS Server

This section will define the relevant portions of the RADIUS message that the server should be looking for, and use the IAS function of a Windows 2003 Server as an example.

RADIUS Attribute Value Pairs (AVPs)

The RADIUS authentication request will contain several Attribute Value Pairs (AVPs) that facilitate the required functions of authentication. They allow the authentication method defined within the RADIUS server to be specific enough to match only on traffic from this client (or class of clients). If the RADIUS server supports logging at high level of verbosity, they contain information about where the client is originating from for logging purposes. The AVPs that the device will send are:

- Username
  - Contains the unencrypted username attempting to authenticate.
- User-Password
  - Contains the encrypted password associated with this authentication attempt.
  - Encrypted using the Pre-Shared Key.
- NAS-Port
  - Indicates the physical port on the unit the user attempting to authenticate is connected to.
- Calling-Station-Id
  - Indicates the IP of the user that is attempting to connect if connecting via IP or "CONSOLE 0/1/2" if connecting via console or dial-in modem.
- Service-Type
  - Set to "Login-User" for login attempts.
  - Set to "Administrative-User" for enable attempts.

- NAS-IP-Address
  - Indicates the primary IP of the interface that the RADIUS request packet is sourced from.

Configuring the RADIUS Server

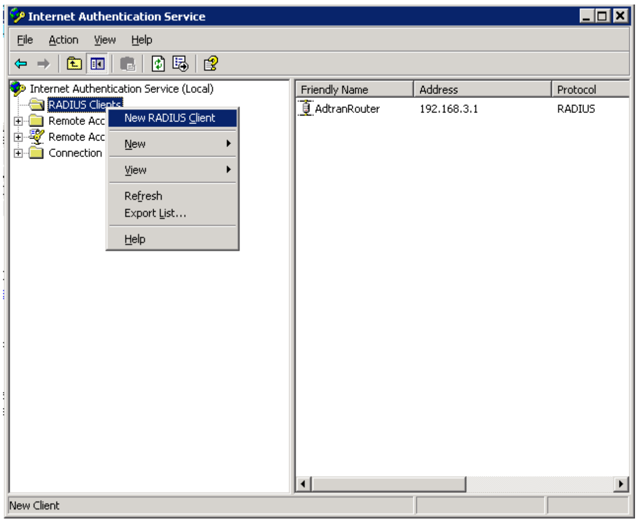
The RADIUS server, using Windows Server 2003's IAS as an example, can specify multiple dependencies that must match before a particular policy is allowed to be used to authenticate a request. It is recommended that these be used to protect the RADIUS server from client authentication requests from unauthorized sources, or to ensure that each RADIUS client has the correct policy applied to it if there are multiple devices sending authentication requests. Examples of such client groups are Device Administrators, Wireless Clients, Port-Authentication, and VPN Clients.

NOTE: *Windows IAS functionality and configuration style may change. The procedures described within this document are only used as an example. ADTRAN is not responsible for configuring the RADIUS server, and will not support the RADIUS server should it be found to be the source of any errors in the authentication process. This article will only cover the Netvanta-specific configuration options within IAS; there may be further configuration on the server required to utilize IAS in this manner.*

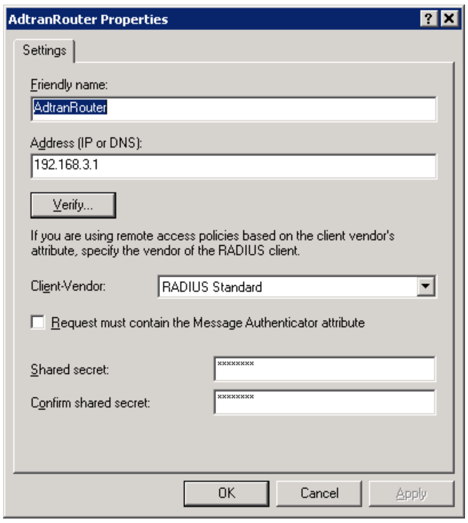
For further information, please refer to the Microsoft KB article on IAS, which can be accessed here:

http://technet.microsoft.com/en-us/library/cc738432(WS.10).aspx

The first step to be completed in most RADIUS servers is to define the RADIUS client device, which involves specifying the Pre-Shared Key and the IP address it will be coming from. This will allow the RADIUS server to receive messages from this RADIUS client. In IAS, it is done in the following manner:



Friendly Name	Address	Protocol
AdtranRouter	192.168.3.1	RADIUS



Friendly name: AdtranRouter

Address (IP or DNS): 192.168.3.1

If you are using remote access policies based on the client vendor's attribute, specify the vendor of the RADIUS client.

Client-Vendor: RADIUS Standard

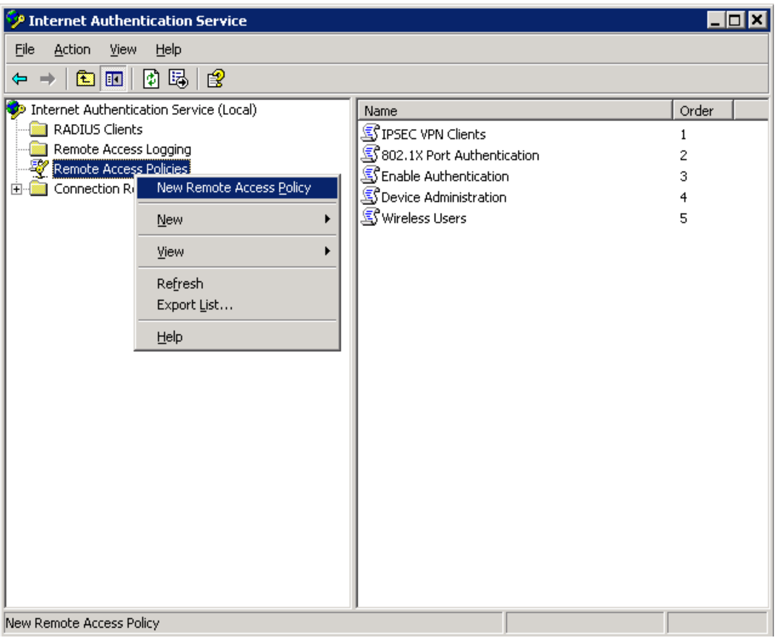
□ Request must contain the Message Authenticator attribute

Shared secret: ********

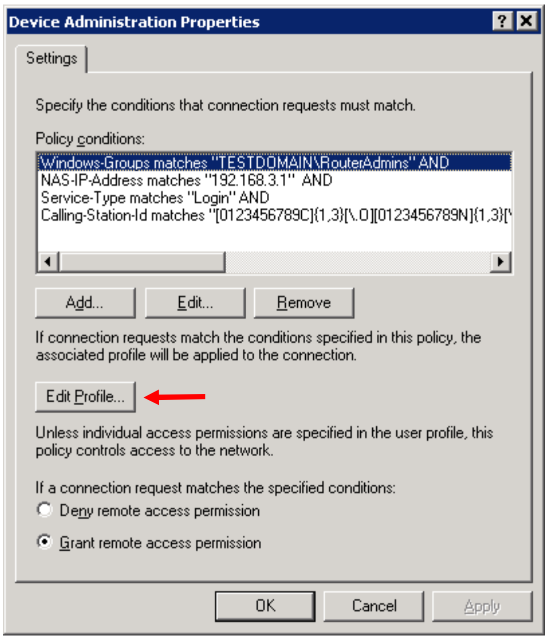
Confirm shared secret: ********

The next step is to create the policy that will process the request, ensure that it matches the required AVPs for this connection type, and permit or deny the request.

The RADIUS server will need to define the AVPs that will remain static within all authentication attempts from this RADIUS client & change the authentication process to allow PAP authentication without negotiation. In IAS, it is done in the following manner:



Name	Order
IPSEC VPN Clients	1
802.1X Port Authentication	2
Enable Authentication	3
Device Administration	4
Wireless Users	5



Specify the conditions that connection requests must match.

Policy conditions:

```
Windows-Groups matches "TESTDOMAIN\RouterAdmins" AND
NAS-IP-Address matches "192.168.3.1" AND
Service-Type matches "Login" AND
Calling-Station-Id matches "[0123456789C]{1,3}[\.0][0123456789N]{1,3}[\
```

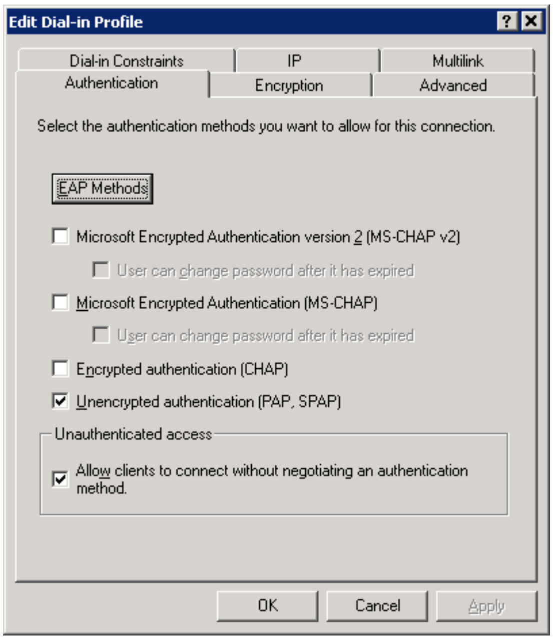
If connection requests match the conditions specified in this policy, the associated profile will be applied to the connection.

Unless individual access permissions are specified in the user profile, this policy controls access to the network.

If a connection request matches the specified conditions:

○ Deny remote access permission

● Grant remote access permission



Select the authentication methods you want to allow for this connection.

□ Microsoft Encrypted Authentication version 2 (MS-CHAP v2)

□ User can change password after it has expired

□ Microsoft Encrypted Authentication (MS-CHAP)

□ User can change password after it has expired

□ Encrypted authentication (CHAP)

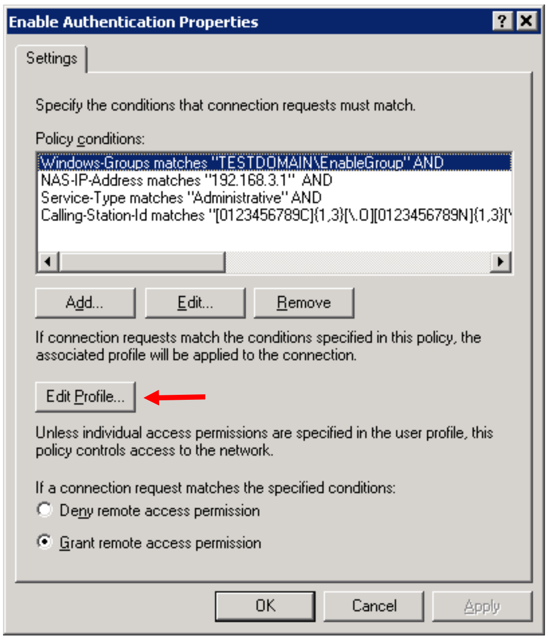
☑ Unencrypted authentication (PAP, SPAP)

Unauthenticated access

☑ Allow clients to connect without negotiating an authentication method.

To further secure the Enable username & password, and to guard against it potentially being used for login purposes, it is recommended that the Enable user be removed from the Domain Users group and placed in its own group within Active Directory. If this step is not taken, it is possible that an attacker could break the enable password, log into the unit using the enable username and password, then enter enable mode using the same password. If the enable username cannot be used for login purposes, two passwords must be obtained to enter the enable mode.

The Enable authentication will be handled by a unique policy that will match "Administrative" rather than "Login" *Service-Type*, and will reference the new Enable group, as shown:



Enable Authentication Properties

Settings

Specify the conditions that connection requests must match.

Policy conditions:

```
Windows-Groups matches "TESTDOMAIN\EnableGroup" AND
NAS-IP-Address matches "192.168.3.1" AND
Service-Type matches "Administrative" AND
Calling-Station-Id matches "[0123456789C]{1,3}[\.O][0123456789N]{1,3}[\
```

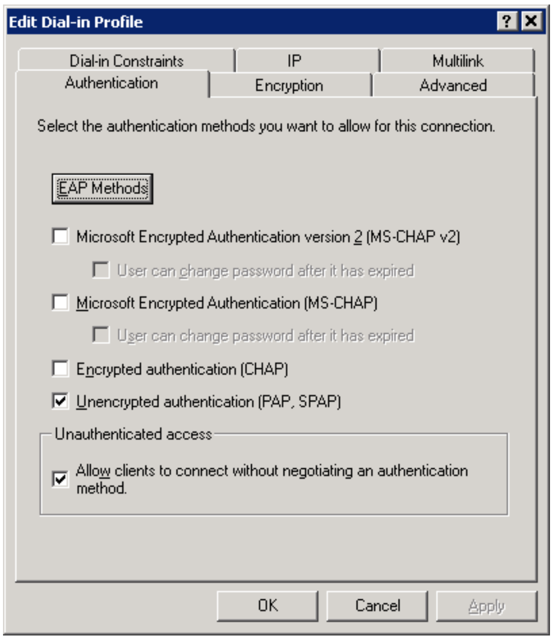
If connection requests match the conditions specified in this policy, the associated profile will be applied to the connection.

Unless individual access permissions are specified in the user profile, this policy controls access to the network.

If a connection request matches the specified conditions:

○ Deny remote access permission

● Grant remote access permission



Edit Dial-in Profile

Select the authentication methods you want to allow for this connection.

□ Microsoft Encrypted Authentication version 2 (MS-CHAP v2)

□ User can change password after it has expired

□ Microsoft Encrypted Authentication (MS-CHAP)

□ User can change password after it has expired

□ Encrypted authentication (CHAP)

☑ Unencrypted authentication (PAP, SPAP)

Unauthenticated access

☑ Allow clients to connect without negotiating an authentication method.

NOTE: *The Calling-Station-Id will always be an IP address if accessed via Telnet, SSH, HTTP or HTTPS; if the device is accessed via console or dial-in modem it will be CONSOLE 0/1/2. This is true for enable authentication as well from the same sources. The string*

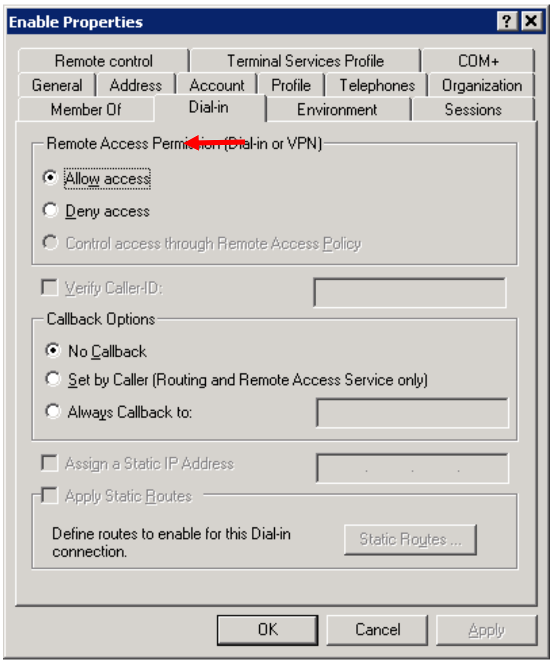
```
[0123456789C]{1,3}[\.O][0123456789N]{1,3}[\.S][01234567890L
E]{1,3}[\.\s][0123456789]{1,3}
```

will ensure that IAS will only match IPv4 addresses & the CONSOLE 0/1/2 string. This is important because the VPN Client RADIUS Requests will also use the "Service-Type" of "Login", but will send a Calling-Station-Id of "XAUTH 100". This will ensure that the Administrator login policy will not process VPN Client requests. Other RADIUS servers may differ in the match string required.

NOTE: *If <u>BOTH</u> the console port and the enable password will <u>NOT</u> be using RADIUS authentication, then the following, much simpler, string can be used to match on IPv4 addresses only:*

```
.+\..+ \..+\..+
```

NOTE: *It may also be required that the users referenced by the policy will need to have "Remote Access Permission", which is enabled on a per-user basis under the "Active Directory Users and Computers" console. An example for enabling this feature is shown:*



Remote Access Permission (Dial-in or VPN)

● Allow access

○ Deny access

○ Control access through Remote Access Policy

□ Verify Caller-ID:

Callback Options

● No Callback

○ Set by Caller (Routing and Remote Access Service only)

○ Always Callback to:

□ Assign a Static IP Address

□ Apply Static Routes

Define routes to enable for this Dial-in connection.

Troubleshooting

This section will describe the relevant debug procedures involved when determining any issues with the AAA or RADIUS configurations. The commands that will be used are:

- debug aaa
- debug radius

A successful authentication attempt would be similar to the following:

```
AAA: New Session on portal 'SSH 1 (<Client IP>:<Port>)'.
AAA: Session using AUTHENTICATION list 'LoginUseRadiusLocal'.
RADIUS AUTHENTICATION: Sending packet to <Server IP> (1812).
RADIUS AUTHENTICATION: Waiting on response from server
RADIUS AUTHENTICATION: Receiving from RADIUS socket
RADIUS AUTHENTICATION: Response received from server (<Server
IP>)
RADIUS AUTHENTICATION: Received response from <Server IP>.
AAA: RADIUS authentication passed.
AAA: Authorization passed. Entering command level 1.
```

An unsuccessful authentication attempt would be similar to the following:

```
AAA: New Session on portal 'SSH 1 (<Client IP>:<Port>)'.
AAA: Session using AUTHENTICATION list 'LoginUseRadiusLocal'.
RADIUS AUTHENTICATION: Sending packet to <Server IP> (1812).
RADIUS AUTHENTICATION: Waiting on response from server
RADIUS AUTHENTICATION: Receiving from RADIUS socket
RADIUS AUTHENTICATION: Response received from server (<Server
IP>)
RADIUS AUTHENTICATION: Received response from <Server IP>.
AAA: RADIUS authentication failed.
AAA: Closing Session on portal 'SSH 1 (<Client IP>:<Port>)'.
```

A failed communication with the RADIUS server and subsequent successful fallback authentication to the Local-User List would be similar to the following:

```
AAA: New Session on portal 'SSH 1 (<Client IP>:<Port>)'.
AAA: Session using AUTHENTICATION list 'LoginUseRadiusLocal'.
RADIUS AUTHENTICATION: Sending packet to <Server IP> (1812).
RADIUS AUTHENTICATION: Waiting on response from server
RADIUS AUTHENTICATION: Receive timed out
RADIUS AUTHENTICATION: Retransmitting packet to <Server IP>
(1812).
RADIUS AUTHENTICATION: Waiting on response from server
RADIUS AUTHENTICATION: Receive timed out
RADIUS AUTHENTICATION: Retransmitting packet to <Server IP>
(1812).
RADIUS AUTHENTICATION: Waiting on response from server
RADIUS AUTHENTICATION: Receive timed out
RADIUS AUTHENTICATION: Retransmitting packet to <Server IP>
(1812).
RADIUS AUTHENTICATION: Waiting on response from server
RADIUS AUTHENTICATION: Receive timed out
RADIUS AUTHENTICATION: Marking server <Server IP> (1812) (1813)
as dead.
RADIUS AUTHENTICATION: Failed to get response from server(s).
AAA: No answer from the RADIUS server(s).
AAA: Authorization passed. Entering command level 1.
```

A failed communication with the RADIUS server and subsequent unsuccessful fallback authentication to the Local-User List would be similar to the following:

```
AAA: New Session on portal 'SSH 1 (<Client IP>:<Port>)'.
AAA: Session using AUTHENTICATION list 'LoginUseRadiusLocal'.
RADIUS AUTHENTICATION: Sending packet to <Server IP> (1812).
RADIUS AUTHENTICATION: Waiting on response from server
RADIUS AUTHENTICATION: Receive timed out
RADIUS AUTHENTICATION: Retransmitting packet to <Server IP>
(1812).
RADIUS AUTHENTICATION: Waiting on response from server
RADIUS AUTHENTICATION: Receive timed out
RADIUS AUTHENTICATION: Retransmitting packet to <Server IP>
(1812).
RADIUS AUTHENTICATION: Waiting on response from server
RADIUS AUTHENTICATION: Receive timed out
RADIUS AUTHENTICATION: Retransmitting packet to <Server IP>
(1812).
RADIUS AUTHENTICATION: Waiting on response from server
RADIUS AUTHENTICATION: Receive timed out
RADIUS AUTHENTICATION: Marking server <Server IP> (1812) (1813)
as dead.
RADIUS AUTHENTICATION: Failed to get response from server(s).
AAA: Closing Session on portal 'SSH 1 (76.164.174.99:4228)'.
```

DISCLAIMER

ADTRAN provides the foregoing application description solely for the reader's consideration and study, and without any representation or suggestion that the foregoing application is or may be free from claims of third parties for infringement of intellectual property rights, including but not limited to, direct and contributory infringement as well as for active inducement to infringe. In addition, the reader's attention is drawn to the following disclaimer with regard to the reader's use of the foregoing material in products and/or systems. That is:

ADTRAN SPECIFICALLY DISCLAIMS ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO, MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT SHALL ADTRAN BE LIABLE FOR ANY LOSS OR DAMAGE, AND FOR PERSONAL INJURY, INCLUDING BUT NOT LIMITED TO, COMPENSATORY, SPECIAL, INCIDENTAL, CONSEQUENTIAL, OR OTHER DAMAGES.