The ArubaOS software allows you to use an external authentication server or the controller internal user

database to authenticate clients who need to access the wireless network.

Important Points to Remember

􀁺 In order for an external authentication server to process requests from the Aruba controller, you must

configure the server to recognize the controller. Refer to the vendor documentation for information on

configuring the authentication server.

􀁺 Instructions on how to configure Microsoft’s IAS and Active Directory can be viewed at:

Microsoft’s IAS

http://technet2.microsoft.com/windowsserver/en/technologies/ias.mspx

Active Directory

http://technet2.microsoft.com/windowsserver/en/technologies/featured/ad/default.mspx

This chapter describes the following topics:

􀁺 “Servers and Server Groups” on page 247

􀁺 “Configuring Servers” on page 248

􀁺 “Configuring the Internal Database” on page 252

􀁺 “Configuring Server Groups” on page 256

Servers and Server Groups

ArubaOS supports the following external authentication servers:

􀁺 RADIUS (Remote Authentication Dial-In User Service)

􀁺 LDAP (Lightweight Directory Access Protocol)

􀁺 TACACS+ (Terminal Access controller Access Control System)

􀁺 Windows (For stateful NTLM authentication)

Additionally, you can use the controller’s internal database to authenticate users. You create entries in the

database for users and their passwords and default role.

You can create *groups* of servers for specific types of authentication. For example, you can specify one or

more RADIUS servers to be used for 802.1x authentication. The list of servers in a server group is an

ordered list. This means that the first server in the list is always used unless it is unavailable, in which case

the next server in the list is used. You can configure servers of different types in one group — for example,

you can include the internal database as a backup to a RADIUS server.

Figure 39 graphically represents a server group named “Radii” that consists of two RADIUS servers, Radius-

1 and Radius-2. The server group is assigned to the server group for 802.1x authentication.

248 | Authentication Servers ArubaOS 5.0 | User Guide

Figure 39 *Server Group*

Server names are unique. You can configure the same server in multiple server groups. You must configure

the server before you can add it to a server group.

You can also include conditions for server-derived user roles or VLANs in the server group configuration.

The server derivation rules apply to all servers in the group.

Configuring Servers

This section describes how to configure RADIUS, LDAP, TACACS+ and Windows external authentication

servers and the internal database on the controller.

Configuring a RADIUS Server

Table 42 describes the parameters you configure for a RADIUS server.

RADIUS-1

RADIUS-2

Server

Group Radii

802.1x

Server Group

N O T E

If you are using the controller’s internal database for user authentication, use the predefined “Internal” server group.

Table 42 *RADIUS Server Configuration Parameters*

Parameter Description

Host IP address of the authentication server.

Default: N/A

Key Shared secret between the controller and the authentication server. The maximum length is

128 characters.

Default: N/A

Authentication Port Authentication port on the server.

Default: 1812

Accounting Port Accounting port on the server

Default: 1813

Retransmits Maximum number of retries sent to the server by the controller before the server is marked

as down.

Default: 3

Timeout Maximum time, in seconds, that the controller waits before timing out the request and

resending it.

Default: 5 seconds

ArubaOS 5.0 | User Guide Authentication Servers | 249

Using the WebUI to configure a RADIUS server

1. Navigate to the **Configuration > Security > Authentication > Servers** page.

2. Select **Radius Server** to display the Radius Server List.

3. To configure a RADIUS server, enter the name for the server and click **Add**.

4. Select the name to configure server parameters. Enter parameters as described in Table 42. Select the

**Mode** checkbox to activate the authentication server.

5. Click **Apply** to apply the configuration.

Using the CLI to configure a RADIUS server

aaa authentication-server radius <*name>*

host <*ipaddr*>

key <*key*>

enable

RADIUS Server Authentication Codes

A configured RADIUS server will return the following standard response codes.

NAS ID Network Access Server (NAS) identifier to use in RADIUS packets.

Default: N/A

NAS IP NAS IP address to send in RADIUS packets.

You can configure a “global” NAS IP address that the controller uses for communications

with all RADIUS servers. If you do not configure a server-specific NAS IP, the global NAS IP

is used. To set the global NAS IP in the WebUI, navigate to the Configuration > Security >

Authentication > Advanced page. To set the global NAS IP in the CLI, enter the ip radius

nas-ip *ipaddr* command.

Default: N/A

Use MD5 Use MD5 hash of cleartext password.

Default: disabled

Mode Enables or disables the server.

Default: enabled

Table 42 *RADIUS Server Configuration Parameters (Continued)*

Parameter Description

N O T E

The configuration does not take effect until you perform this step.

Table 43 *RADIUS Authentication Response Codes*

Code Description

0 Authentication OK.

1 Authentication failed—user/password combination not correct.

2 Authentication request timed out—No response from server.

3 Internal authentication error.

250 | Authentication Servers ArubaOS 5.0 | User Guide

Configuring an LDAP Server

Table 44 describes the parameters you configure for an LDAP server.

Using the WebUI to configure an LDAP server

1. Navigate to the **Configuration > Security > Authentication > Servers** page.

2. Select **LDAP Server** to display the LDAP Server List.

3. To configure an LDAP server, enter the name for the server and click **Add**.

4. Select the name to configure server parameters. Enter parameters as described in Table 44. Select the

**Mode** checkbox to activate the authentication server.

4 Bad Response from RADIUS server. Verify shared secret is correct.

5 No RADIUS authentication server is configured.

6 Challenge from server. (This does not necessarily indicate an error condition.)

Table 44 *LDAP Server Configuration Parameters*

Parameter Description

Host IP address of the LDAP server.

Default: N/A

Admin-DN Distinguished name for the admin user who has read/search privileges across all the

entries in the LDAP database (the user need not have write privileges but the user should

be able to search the database, and read attributes of other users in the database).

Admin Password Password for the admin user.

Default: N/A

Allow Clear-Text Allows clear-text (unencrypted) communication with the LDAP server.

Default: disabled

Authentication Port Port number used for authentication.

Default: 389

Base-DN Distinguished Name of the node which contains the entire user database to use.

Default: N/A

Filter Filter that should be applied to search of the user in the LDAP database (default filter

string is: i(objectclass=\*)i ).

Default: N/A

Key Attribute Attribute that should be used as a key in search for the LDAP server. For Active Directory,

the value is sAMAccountName.

Default: sAMAccountName

Timeout Timeout period of a LDAP request, in seconds.

Default: 20 seconds

Mode Enables or disables the server.

Default: enabled

Table 43 *RADIUS Authentication Response Codes*

Code Description

ArubaOS 5.0 | User Guide Authentication Servers | 251

5. Click **Apply** to apply the configuration.

Using the CLI to configure an LDAP server

aaa authentication-server ldap <*name>*

host <ipaddr>

*(enter parameters as described in* Table 44*)*

enable

Configuring a TACACS+ Server

Table 45 defines the TACACS+ server parameters.

Using the WebUI to configure a TACACS+ server

1. Navigate to the **Configuration > Security > Authentication > Servers** page.

2. Select **TACACS Server** to display the TACACS Server List.

3. To configure a TACACS+ server, enter the name for the server and click **Add**.

4. Select the name to configure server parameters. Enter parameters as described in Table 45. Select the

**Mode** checkbox to activate the authentication server.

5. Click **Apply** to apply the configuration.

Using the CLI to configure a TACACS+ server

aaa authentication-server tacacs <*name>*

host <ipaddr>

key <key>

enable

N O T E

The configuration does not take effect until you perform this step.

Table 45 *TACACS+ Server Configuration Parameters*

Parameter Description

Host IP address of the server.

Default: N/A

Key Shared secret to authenticate communication between the TACACS+ client and server.

Default: N/A

TCP Port TCP port used by server.

Default: 49

Retransmits Maximum number of times a request is retried.

Default: 3

Timeout Timeout period for TACACS+ requests, in seconds.

Default: 20 seconds

Mode Enables or disables the server.

Default: enabled

N O T E

The configuration does not take effect until you perform this step.

252 | Authentication Servers ArubaOS 5.0 | User Guide

Configuring a Windows Server

Table 46 defines parameters for a Windows server used for stateful NTLM authentication.

Using the WebUI to configure a Windows server

1. Navigate to the **Configuration > Security > Authentication > Servers** page.

2. Select **Windows Server** to display the Windows Server List.

3. To configure a Windows server, enter the name for the server and click **Add**.

4. Select the name of the server to configure its parameters. Enter the parameters as described in Table 46.

5. Select the **Mode** checkbox to activate the authentication server.

6. Click **Apply** to apply the configuration.

Using the CLI to configure a Windows server

aaa authentication-server windows <*windows-server-name>*

host <ipaddr>

enable

Configuring the Internal Database

You can create entries, in the controller’s internal database, to use to authenticate clients. The internal

database contains a list of clients along with the password and default role for each client. When you

configure the internal database as an authentication server, client information in incoming authentication

requests is checked against the internal database.

By default, the internal database in the master controller is used for authentication. You can choose to use

the internal database in a local controller by entering the CLI command **aaa authentication-server**

**internal use-local-switch**. If you use the internal database in a local controller, you need to add clients on

the local controller.

.

Table 47 defines the required and optional parameters used in the internal database.

Table 46 *Windows Server Configuration Parameters*

Parameter Description

Host IP address of the server.

Default: N/A

Mode Enables or disables the server.

Default: enabled

N O T E

The configuration does not take effect until you perform this step.

Table 47 *Internal Database Configuration Parameters*

Parameters Description

User Name (Required) Enter a user name or select Generate to automatically generate a user name. An

entered username can be up to 64 characters in length.

ArubaOS 5.0 | User Guide Authentication Servers | 253

Using the WebUI to configure users in the internal database

1. Navigate to the **Configuration > Security > Authentication > Servers >** page.

2. Select **Internal DB**.

3. Click **Add User** in the **Users** section. The user configuration page displays.

4. Enter the information for the client, as described in the table above.

5. Click **Enabled** to activate this entry on creation.

6. Click **Apply** to apply the configuration. The configuration does not take effect until you perform this

step

7. At the Servers page, click **Apply**.

Using the CLI to configure users in the internal database

Enter the following command in enable mode:

local-userdb add {generate-username|username <*name>*} {generate-password|password

<*password>*}

RAP Static Inner IP Address

The RAP static inner IP address feature assigns a static inner IP address to a remote access point (RAP). A

new *IP address* parameter is added to the existing configuration commands: local-userdb add, localuserdb

modify, local-userdb-ap add, and local-userdb-ap modify**.**

Password (Required) Enter a password or select Generate to automatically generate a password

string. An entered password must be a minimum of 6 characters and can be up to 128

characters in length.

Role Role for the client.

In order for this role to be assigned to a client, you need to configure a server derivation

rule, as described in “Configuring Server-Derivation Rules” on page 260. (A user role

assigned through a server-derivation rule takes precedence over the default role configured

for an authentication method.)

E-mail (Optional) E-mail address of the client.

Enabled Select this checkbox to enable the user as soon as the user entry is created.

Expiration Select one of the following options:

􀁺 Entry does not expire: No expiration on user entry

􀁺 Set Expiry time (mins): Enter the number of minutes the user will be authenticated

before their user entry expires.

􀁺 Set Expiry Date (mm/dd/yyyy) Expiry Time (hh:mm): To select a specific expiration date

and time, enter the expiration date in mm/dd/yyyy format, and the expiration time in

hh:mm format.

Static Inner IP

Address (for RAPs

only)

Assign a static inner IP address to a Remote AP. If this database entry is not for a remote

AP, leave this field empty.

Table 47 *Internal Database Configuration Parameters (Continued)*

Parameters Description

N O T E

The Internal DB Maintenance window also includes a Guest User Page feature that allows you to create user entries

for guests only. For details on creating guest users, see “Guest Provisioning User Tasks” on page 542.

254 | Authentication Servers ArubaOS 5.0 | User Guide

Using the WebUI

To view *IP address* parameter in the local database, navigate to the **Configuration** > **Security** >

**Authentication** > **Servers** > **Internal DB** page.

Figure 40 *IP-Address parameter in the local database*

To view IP-address parameter in the RAP Whitelist, navigate to the **Wireless** > **AP Installation** > **RAP**

**Whitelist** page.

Figure 41 *IP-Address parameter in the RAP Whitelist*

Using CLI

local-userdb add {generate-username|username <name>} {generate-password|password

<password>} {remote-ip<remote-ip>}

local-userdb modify {username < name>} {remote-ip<remote-ip>}

local-userdb-ap add {mac-address <address>} {ap-group|<ap\_grup>} {remote-ip<remoteip>}

local-userdb-ap modify { mac-address <address>} {remote-ip<remote-ip>}

The output of **show local-userdb** command:

N O T E

You cannot configure the IP-Address parameter by using the WebUI.

(host) #show local-userdb

User Summary

------------

Name Pwd Role E-Mail Enabled Expiry Status Sponsor-Name Remote-IP Grantor-Name

---- --- ---- ------ ------- ------ ------ ------------ --------- ------------

John \*\*\* default-vpn-role john@example.com Yes Active 0.0.0.0 admin

user1 \*\*\* default-vpn-role Yes Active 0.0.0.0 admin

Sam \*\*\* default-vpn-role Yes Active 0.0.0.0 admin

ArubaOS 5.0 | User Guide Authentication Servers | 255

The output of **show local-userdb-ap** command:

Managing Internal Database Files

ArubaOS allows you to import and export tables of user information to and from the internal database.

These files should not be edited once they are exported. ArubaOS only supports the importing of database

files that were created during the export process. Note that importing a file into the internal database

overwrite and removes all existing entries.

Using the WebUI to export files from the internal database

1. Navigate to the **Configuration > Security > Authentication > Servers >** page.

2. Select **Internal DB**.

3. Click **Export** in the **Internal DB Maintenance** section. A popup window opens.

4. Enter the name of the file you want to export

5. Click **OK**.

Using the WebUI to export files from the internal database

1. Navigate to the **Configuration > Security > Authentication > Servers >** page.

2. Select **Internal DB**.

3. Click **Import** in the **Internal DB Maintenance** section. A popup window opens.

4. Enter the name of the file you want to import

5. Click **OK**.

Using the CLI to export and import users in the internal database

Enter the following command in enable mode:

local-userdb export <filename>

local-userdb import <filename>

Internal Database Utilities

The local internal database also includes utilities to clear all users from the database and to restart the

internal database to repair internal errors. Under normal circumstances, neither of these utilities are

necessary.

Using the WebUI to delete all users from the internal database

Issue this command to remove users from the internal database after you have moved your user database

from the controller’s internal server to an external server.

1. Navigate to the **Configuration > Security > Authentication > Servers >** page.

2. Select **Internal DB**.

3. Click **Delete All Users** in the **Internal DB Maintenance** section. A popup window open and asks you

to confirm that you want to remove all users.

(host) #show local-userdb-ap

AP-entry Details

----------------

Name AP-Group AP-Name Full-Name Auth-Uname Rvok-txt AP\_Auth Descrp Date-Added En Rem-IP

---- -------- ------- --------- ---------- -------- ----------- ------ ---------- --- -------

MAC-ADD CP\_TEST AP-125-Port-2 test Provisioned wq Fri Nov 27 2009 Yes 0.0.0.0

MAC-ADD CP\_TEST AP-rap5-port-18 John Provisioned desc Mon Nov 30 2009 Yes 0.0.0.0

256 | Authentication Servers ArubaOS 5.0 | User Guide

4. Click **OK**.

Using the WebUI to repair the internal database

Use this utility under the supervision of Aruba technical support to recreate the internal database. This may

clear internal database errors, but will also remove all information from the database. Make sure you export

your current user information before you start the repair procedure.

1. Navigate to the **Configuration > Security > Authentication > Servers >** page.

2. Select **Internal DB**.

3. Click **Repair Database** in the **Internal DB Maintenance** section. A popup window open and asks you

to confirm that you want to recreate the database.

4. Click **OK**.

Configuring Server Groups

You can create *groups* of servers for specific types of authentication — for example, you can specify one or

more RADIUS servers to be used for 802.1x authentication. You can configure servers of different types in

one group — for example, you can include the internal database as a backup to a RADIUS server.

Server names are unique. You can configure the same server in more than one server group. The server

must be configured before you can include it in a server group.

Using the WebUI to configure a server group

1. Navigate to the **Configuration > Security > Authentication > Servers** page.

2. Select **Server Group** to display the Server Group list.

3. Enter the name of the new server group and click **Add**.

4. Select the name to configure the server group.

5. Under Servers, click **New** to add a server to the group.

a. Select a server from the drop-down menu and click **Add Server**.

b. Repeat the above step to add other servers to the group.

6. Click **Apply**.

Using the CLI to configure a server group

aaa server-group <*name*>

auth-server <*name>*

Server List Order and Fail-Through

The list of servers in a server group is an ordered list. By default, the first server in the list is always used

unless it is unavailable, in which case the next server in the list is used. You can configure the order of

servers in the server group. In the WebUI, use the up or down arrows to order the servers (the top server is

the first server in the list). In the CLI, use the **position** parameter to specify the relative order of servers in

the list (the lowest value denotes the first server in the list).

As mentioned previously, the first available server in the list is used for authentication. If the server

responds with an authentication failure, there is no further processing for the user or client for which the

authentication request failed. You can optionally enable *fail-through* authentication for the server group so

that if the first server in the list returns an authentication deny, the controller attempts authentication with

the next server in the ordered list. The controller attempts authentication with each server in the list until

either there is a successful authentication or the list of servers in the group is exhausted. This feature is

useful in environments where there are multiple, independent authentication servers; users may fail

authentication on one server but can be authenticated on another server.

ArubaOS 5.0 | User Guide Authentication Servers | 257

Before enabling fail-through authentication, note the following:

􀁺 This feature is not supported for 802.1x authentication with a server group that consists of external EAPcompliant

RADIUS servers. You can, however, use fail-through authentication when the 802.1x

authentication is terminated on the controller (AAA FastConnect).

􀁺 Enabling this feature for a large server group list may cause excess processing load on the controller.

Aruba recommends that you use server selection based on domain matching whenever possible (see

“Dynamic Server Selection” on page 257).

􀁺 Certain servers, such as the RSA RADIUS server, lock out the controller if there are multiple

authentication failures. Therefore you should not enable fail-through authentication with these servers.

In the following example, you create a server group ‘corp-serv’ with two LDAP servers (ldap-1 and ldap-2),

each of which contains a subset of the usernames and passwords used in the network. When fail-through

authentication is enabled, users that fail authentication on the first server in the server list should be

authenticated with the second server.

Using the WebUI to configure fail-through authentication

1. Navigate to the **Configuration > Security > Authentication > Servers** page.

2. Select **LDAP Server** to display the LDAP Server List.

3. Enter ldap-1 for the server name and click **Add**.

4. Enter ldap-2 for the server name and click **Add**.

5. Under the Servers tab, select ldap-1 to configure server parameters. Enter the IP address for the server.

Select the **Mode** checkbox to activate the authentication server. Click **Apply**.

6. Repeat step 5 to configure ldap-2.

7. Display the Server Group list: Under the Servers tab, select **Server Group**.

8. Enter **corp-serv** as the new server group and click **Add**.

9. Select **corp-serv**, under the Server tab, to configure the server group.

10. Select **Fail Through**.

11. Under Servers, click **New** to add a server to the group. Select ldap-1 from the drop-down menu and click

**Add Server**.

12. Repeat step 11 to add ldap-2 to the group.

13. Click **Apply**.

Using the CLI to configure fail-through authentication

aaa authentication-server ldap ldap-1

host 10.1.1.234

aaa authentication-server ldap ldap-2

host 10.2.2.234

aaa server-group corp-serv

auth-server ldap-1 position 1

auth-server ldap-2 position 2

allow-fail-through

Dynamic Server Selection

The controller can dynamically select an authentication server from a server group based on the user

information sent by the client in an authentication request. For example, an authentication request can

include client or user information in one of the following formats:

􀁺 <domain>\<user> — for example, corpnet.com\darwin

258 | Authentication Servers ArubaOS 5.0 | User Guide

􀁺 <user>@<domain> — for example, darwin@corpnet.com

􀁺 host/<pc-name>.<domain> — for example, host/darwin-g.finance.corpnet.com (this format is used with

802.1x machine authentication in Windows environments)

When you configure a server in a server group, you can optionally associate the server with one or more

match rules. A match rule for a server can be one of the following:

􀁺 The server is selected if the client/user information *contains* a specified string.

􀁺 The server is selected if the client/user information *begins* with a specified string.

􀁺 The server is selected if the client/user information *exactly* matches a specified string.

You can configure multiple match rules for the same server. The controller compares the client/user

information with the match rules configured for each server, starting with the first server in the server

group. If a match is found, the controller sends the authentication request to the server with the matching

rule. If no match is found before the end of the server list is reached, an error is returned and no

authentication request for the client/user is sent.

For example, Figure 42 depicts a network consisting of several subdomains in corpnet.com. The server

radius-1 provides 802.1x machine authentication to PC clients in xyz.corpnet.com, sales.corpnet.com, and

hq.corpnet.com. The server radius-2 provides authentication for users in abc.corpnet.com.

Figure 42 *Domain-Based Server Selection Example*

You configure the following rules for servers in the corp-serv server group:

􀁺 radius-1 will be selected if the client information starts with “host/”.

􀁺 radius-2 will be selected if the client information contains “abc.corpnet.com”.

Using the WebUI to configure server selection

1. Navigate to the **Configuration > Security > Authentication > Servers** page.

2. Under the Servers tab, select **Server Group** to display the Server Group list.

3. Enter **corp-serv** for the new server group and click **Add**.

4. Under the Servers tab, select **corp-serv** to configure the server group.

5. Under Servers, click **New** to add the radius-1 server to the group. Select radius-1 from the drop-down

menu.

a. For Match Type, select **Authstring**.

b. For Operator, select **starts-with**.

c. For Match String, enter **host/**.

d. Click **Add Rule >>**.

e. Scroll to the right and click **Add Server**.

host/<pc-name>.xyz.corpnet.com

abc.corpnet.com\<user>

<user>@abc.corpnet.com

radius-1

radius-2

host/<pc-name>.sales.corpnet.com

host/<pc-name>.hq.corpnet.com

ArubaOS 5.0 | User Guide Authentication Servers | 259

6. Under Servers, click **New** to add the radius-2 server to the group. Select radius-2 from the drop-down

menu.

a. For Match Type, select **Authstring**.

b. For Operator, select **contains**.

c. For Match String, enter **abc.corpnet.com**.

d. Click **Add Rule >>**.

e. Scroll to the right and click **Add Server**.

7. Click **Apply**.

Using the CLI to configure server selection

aaa server-group corp-serv

auth-server radius-1 match-authstring starts-with host/ position 1

auth-server radius-2 match-authstring contains abc.corpnet.com position 2

Match FQDN Option

You can also use the “match FQDN” option for a server match rule. With a match FQDN rule, the server is

selected if the <domain> portion of the user information in the formats <domain>\<user> or

<user>@<domain> *exactly* matches a specified string. Note the following caveats when using a match

FQDN rule:

􀁺 This rule does *not* support client information in the host/<pc-name>.<domain> format, so it is not useful

for 802.1x machine authentication.

􀁺 The match FQDN option performs matches on only the <domain> portion of the user information sent in

an authentication request. The match-authstring option (described previously) allows you to match all

or a portion of the user information sent in an authentication request.

Using the WebUI to configure match FQDN option

1. Navigate to the **Configuration > Security > Authentication > Servers** page

2. Under the Servers tab, select **Server Group** to display the Server Group list.

3. Enter **corp-serv** for the new server group and click **Add**.

4. Under the Servers tab, select **corp-serv** to configure the server group.

5. Under Servers, click **New** to add the radius-1 server to the group. Select radius-1 from the drop-down

menu.

a. For Match Type, select **FQDN**.

b. For Match String, enter **corpnet.com**.

c. Click **Add Rule >>**.

d. Scroll to the right and click **Add Server**.

6. Click **Apply**.

Using the CLI to configure match FQDN option

aaa server-group corp-serv

auth-server radius-1 match-fqdn corpnet.com

N O T E

The last server you added to the server group (radius-2) automatically appears as the first server in the list. In this

example, the order of servers is not important. If you need to reorder the server list, scroll to the right and click the

up or down arrow for the appropriate server.

260 | Authentication Servers ArubaOS 5.0 | User Guide

Trimming Domain Information from Requests

Before the controller forwards an authentication request to a specified server, it can truncate the domainspecific

portion of the user information. This is useful when user entries on the authenticating server do not

include domain information. You can specify this option with any server match rule. This option is only

applicable when the user information is sent to the controller in the following formats:

􀁺 <domain>\<user> — the <domain>\ portion is truncated

􀁺 <user>@<domain> — the @<domain> portion is truncated

Using the WebUI to trim domain information

1. Navigate to the **Configuration > Security > Authentication > Servers** page.

2. Select **Server Group** to display the Server Group list.

3. Enter the name of the new server group and click **Add**.

4. Select the name to configure the server group.

5. Under Servers, click **Edit** for a configured server or click **New** to add a server to the group.

􀁺 If editing a configured server, select Trim FQDN, scroll right, and click **Update Server**.

􀁺 If adding a new server, select a server from the drop-down menu, then select Trim FQDN, scroll right,

and click **Add Server**.

6. Click **Apply**.

Using the CLI to trim domain information

aaa server-group corp-serv

auth-server radius-2 match-authstring contains abc.corpnet.com trim-fqdn

Configuring Server-Derivation Rules

When you configure a server group, you can set the VLAN or role for clients based on attributes returned for

the client by the server during authentication. The server derivation rules apply to all servers in the group.

The user role or VLAN assigned through server derivation rules takes precedence over the default role and

VLAN configured for the authentication method.

The server rules are applied based on the first match principle. The first rule that is applicable for the server

and the attribute returned is applied to the client and would be the only rule applied from the server rules.

These rules are applied uniformly across all servers in the server group.

Table 48 describes the server rule parameters you can configure.

N O T E

This option does not support client information sent in the format host/<pc-name>.<domain>

N O T E

The authentication servers must be configured to return the attributes for the clients during

authentication. For instructions on configuring the authentication attributes in a Windows environment

using IAS, refer to the documentation at http://technet2.microsoft.com/windowsserver/en/technologies/

ias.mspx.

ArubaOS 5.0 | User Guide Authentication Servers | 261

Using the WebUI to configure server rules

1. Navigate to the **Configuration > Security > Authentication > Servers** page.

2. Select **Server Group** to display the Server Group list.

3. Enter the name of the new server group and click **Add**.

4. Select the name to configure the server group.

5. Under Servers, click **New** to add a server to the group.

a. Select a server from the drop-down menu and click **Add**.

b. Repeat the above step to add other servers to the group.

6. Under Server Rules, click **New** to add server derivation rules for assigning a user role or VLAN.

a. Enter the attribute.

b. Select the operation from the drop-down menu.

c. Enter the operand.

d. Select Set VLAN or Set Role from the drop-down menu.

e. Enter the value (either user role or VLAN) to be assigned.

f. Click **Add**.

g. Repeat the above steps to add other rules for the server group.

Table 48 *Server Rule Configuration Parameters*

Parameter Description

Role or VLAN The server derivation rules can be for either user role or VLAN assignment. With Role

assignment, a client can be assigned a specific role based on the attributes

returned. In case of VLAN assignment, the client can be placed in a specific VLAN

based on the attributes returned.

Attribute This is the attribute returned by the authentication server that is examined for

*Operation* and *Operand* match.

Operation This is the match method by which the string in *Operand* is matched with the

attribute value returned by the authentication server.

􀁺 contains – The rule is applied if and only if the attribute value contains the string

in parameter *Operand.*

􀁺 starts-with – The rule is applied if and only if the attribute value returned starts

with the string in parameter *Operand.*

􀁺 ends-with – The rule is applied if and only if the attribute value returned ends with

the string in parameter *Operand.*

􀁺 equals – The rule is applied if and only if the attribute value returned equals the

string in parameter *Operand.*

􀁺 not-equals – The rule is applied if and only if the attribute value returned is not

equal to the string in parameter *Operand.*

􀁺 value-of – This is a special condition. What this implies is that the role or VLAN is

set to the value of the attribute returned. For this to be successful, the role and

the VLAN ID returned as the value of the attribute selected must be already

configured on the controller when the rule is applied.

Operand This is the string to which the value of the returned attribute is matched.

Value The user role or the VLAN applied to the client when the rule is matched.

position Position of the condition rule. Rules are applied based on the first match principle. 1

is the top.

Default: bottom

262 | Authentication Servers ArubaOS 5.0 | User Guide

7. Click **Apply**.

Using the CLI to configure server rules

aaa server-group <*name*>

auth-server <*name>*

set {role|vlan} condition <*condition>* set-value {<role>|<vlan>}

[position *number*]

Configuring a Role Derivation Rule for the Internal Database

When you add a user entry in the controller’s internal database, you can optionally specify a user role (see

“Configuring the Internal Database” on page 252). In order for the role specified in the internal database

entry to be assigned to the authenticated client, you must configure a server derivation rule as shown in the

following sections:

Using the WebUI to configure a server rule for the internal database

1. Navigate to the **Configuration > Security > Authentication > Servers** page.

2. Select **Server Group** to display the Server Group list.

3. Select the **internal** server group.

4. Under Server Rules, click **New** to add a server derivation rule.

a. For Condition, enter Role.

b. Select value-of from the drop-down menu.

c. Select Set Role from the drop-down menu.

d. Click **Add**.

5. Click **Apply**.

Using the CLI to configure a server rule for the internal database:

aaa server-group internal

set role condition Role value-of

Assigning Server Groups

You can create server groups for the following purposes:

􀁺 user authentication

􀁺 management authentication

􀁺 accounting

You can configure all types of servers for user and management authentication (see Table 49). Accounting

is only supported with RADIUS and TACACS+ servers when RADIUS or TACACS+ is used for

authentication.

Table 49 *Server Types and Purposes*

RADIUS TACACS+ LDAP Internal Database

User authentication Yes Yes Yes Yes

Management authentication Yes Yes Yes Yes

Accounting Yes Yes No No

ArubaOS 5.0 | User Guide Authentication Servers | 263

User Authentication

For information about assigning a server group for user authentication, see the configuration chapter for

the authentication method.

Management Authentication

Users who need to access the controller to monitor, manage, or configure the Aruba user-centric network

can be authenticated with RADIUS, TACACS+, or LDAP servers or the internal database.

Using the WebUI to assign a server group for management authentication

1. Navigate to the **Configuration > Management > Administration** page.

2. Under the Management Authentication Servers section, select the Server Group.

3. Click **Apply**.

Using the CLI to assign a server group for management authentication

aaa authentication mgmt

server-group <*group>*

Accounting

You can configure accounting for RADIUS and TACACS+ server groups.

RADIUS Accounting

RADIUS accounting allows user activity and statistics to be reported from the controller to RADIUS

servers. RADIUS accounting works as follows:

1. The controller generates an Accounting Start packet when a user logs in. The code field of transmitted

RADIUS packet is set to 4 (Accounting-Request). Note that sensitive information, such user passwords,

are not sent to the accounting server. The RADIUS server sends an acknowledgement of the packet.

2. The controller sends an Accounting Stop packet when a user logs off; the packet information includes

various statistics such as elapsed time, input and output bytes and packets. The RADIUS server sends an

acknowledgement of the packet.

The following is the list of attributes that the controller can send to a RADIUS accounting server:

􀁺 Acct-Status-Type: This attribute marks the beginning or end of accounting record for a user. Currently,

possible values include Start and Stop.

􀁺 User-Name: Name of user.

􀁺 Acct-Session-Id: A unique identifier to facilitate matching of accounting records for a user. It is derived

from the user name, IP address and MAC address. This is set in all accounting packets.

􀁺 Acct-Authentic: This indicates how the user was authenticated. Current values are 1 (RADIUS), 2 (Local)

and 3 (LDAP).

􀁺 Acct-Session-Time: The elapsed time, in seconds, that the client was logged in to the controller. This is

only sent in Accounting-Request records where the Acct-Status-Type is Stop.

N O T E

Only user record attributes are returned upon a successful authentication. Therefore, to derive a different

management role other than the default mgmt auth role, set the server derivation rule based on the user attributes.

N O T E

RADIUS or TACACS+ accounting is only supported when RADIUS or TACACS+ is used for authentication.

264 | Authentication Servers ArubaOS 5.0 | User Guide

􀁺 Acct-Terminate-Cause: Indicates how the session was terminated and is sent in Accounting-Request

records where the Acct-Status-Type is Stop. Possible values are:

1: User logged off

4: Idle Timeout

5: Session Timeout. Maximum session length timer expired.

7: Admin Reboot: Administrator is ending service, for example prior to rebooting the controller.

􀁺 NAS-Identifier: This is set in the RADIUS server configuration.

􀁺 NAS-IP-Address: IP address of the master controller. You can configure a “global” NAS IP address: in the

WebUI, navigate to the **Configuration > Security > Authentication > Advanced** page; in the CLI, use

the **ip radius nas-ip** command.

􀁺 NAS-Port: Physical or virtual port (tunnel) number through which the user traffic is entering the

controller.

􀁺 NAS-Port-Type: Type of port used in the connection. This is set to one of the following:

􀂄 5: admin login

􀂄 15: wired user type

􀂄 19: wireless user

􀁺 Framed-IP-Address: IP address of the user.

􀁺 Calling-Station-ID: MAC address of the user.

􀁺 Called-station-ID: MAC address of the controller.

The following attributes are sent in Accounting-Request packets when Acct-Status-Type value is Start:

􀁺 Acct-Status-Type

􀁺 User-Name

􀁺 NAS-IP-Address

􀁺 NAS-Port

􀁺 NAS-Port-Type

􀁺 NAS-Identifier

􀁺 Framed-IP-Address

􀁺 Calling-Station-ID

􀁺 Called-station-ID

􀁺 Acct-Session-Id

􀁺 Acct-Authentic

The following attributes are sent in Accounting-Request packets when Acct-Status-Type value is Stop:

􀁺 Acct-Status-Type

􀁺 User-Name

􀁺 NAS-IP-Address

􀁺 NAS-Port

􀁺 NAS-Port-Type

􀁺 NAS-Identifier

􀁺 Framed-IP-Address

􀁺 Calling-Station-ID

􀁺 Called-station-ID

􀁺 Acct-Session-Id

ArubaOS 5.0 | User Guide Authentication Servers | 265

􀁺 Acct-Authentic

􀁺 Terminate-Cause

􀁺 Acct-Session-Time

The following attributes are sent only in Accounting Stop packets (they are not sent in Accounting Start

packets):

􀁺 Acct-Input-Octets

􀁺 Acct-Output-Octets

􀁺 Acct-Input-Packets

􀁺 Acct-Output-Packets

You can use either the WebUI or CLI to assign a server group for RADIUS accounting.

Using the WebUI to assign a server group for RADIUS accounting

1. Navigate to the **Configuration > Security > Authentication > AAA Profiles** page.

2. Select AAA Profile, then select the AAA profile instance.

3. Scroll down and select Radius Accounting Server Group. Select the server group from the drop-down

menu.

You can add additional servers to the group or configure server rules.

4. Click **Apply**.

Using the CLI to assign a server group for RADIUS accounting

aaa profile <*profile>*

radius-accounting <*group>*

TACACS+ Accounting

TACACS+ accounting allows commands issued on the controller to be reported to TACACS+ servers. You

can specify the types of commands that are reported (action, configuration, or show commands) or have all

commands reported.

You can configure TACACS+ accounting only with the CLI:

aaa tacacs-accounting server-group <*group>* command {action|all|configuration|show} mode

{enable|disable}

266 | Authentication Servers ArubaOS 5.0 | User Guide

Configuring Authentication Timers

Table 50 describes the timers you can configure that apply to all clients and servers. These timers can be left

at their default values for most implementations.

Using the WebUI to set an authentication timer

1. Navigate to the **Configuration > Security > Authentication > Advanced** page.

2. Configure the timers as described above.

3. Click **Apply** before moving on to another page or closing the browser window. Failure to do this results

in loss of configuration and you will have to reconfigure the settings.

Using the CLI to set an authentication timer:

aaa timers {dead-time <minutes>|idle-timeout <number>|logon-lifetime <minutes>}

Table 50 *Authentication Timers*

Timer Description

User Idle Timeout Maximum period, in minutes or seconds, after which a client is considered

idle if there is no user traffic from the client.

The timeout period is reset if there is a user traffic. After this timeout period

has elapsed, the controller sends probe packets to the client; if the client

responds to the probe, it is considered active and the User Idle Timeout is

reset (an active client that is not initiating new sessions is not removed). If

the client does not respond to the probe, it is removed from the system. To

prevent clients from timing out, set the value in the field to 0. After entering

the value, select either min for minutes or sec for seconds from the dropdown

list.

Range: 0–255

Default: 5 minutes

Authentication Server

Dead Time

Maximum period, in minutes, that the controller considers an unresponsive

authentication server to be “out of service”.

This timer is only applicable if there are two or more authentication servers

configured on the controller. If there is only one authentication server

configured, the server is never considered out of service and all requests

are sent to the server.

If one or more backup servers are configured and a server is unresponsive,

it is marked as out of service for the dead time; subsequent requests are

sent to the next server on the priority list for the duration of the dead time. If

the server is responsive after the dead time has elapsed, it can take over

servicing requests from a lower-priority server; if the server continues to be

unresponsive, it is marked as down for the dead time.

Range: 0–50

Default: 10 minutes

Logon User Lifetime Maximum time, in minutes, unauthenticated clients are allowed to remain

logged on.

Range: 0–255

Default: 5 minutes

ArubaOS 5.0 | User Guide 802.1x Authentication | 267

Chapter 10

802.1x Authentication

802.1x is an Institute of Electrical and Electronics Engineers (IEEE) standard that provides an

authentication framework for WLANs. 802.1x uses the Extensible Authentication Protocol (EAP) to

exchange messages during the authentication process. The authentication protocols that operate inside the

802.1x framework that are suitable for wireless networks include EAP-Transport Layer Security (EAP-TLS),

Protected EAP (PEAP), and EAP-Tunneled TLS (EAP-TTLS). These protocols allow the network to

authenticate the client while also allowing the client to authenticate the network.

This chapter describes the following topics:

􀁺 “Overview of 802.1x Authentication” on page 267

􀁺 “Configuring 802.1x Authentication” on page 270

􀁺 “Example Configurations” on page 278

􀁺 “Advanced Configuration Options for 802.1x” on page 298

Other types of authentication not discussed in this chapter can be found in the following sections of this

guide:

􀁺 Captive portal authentication: “Configuring Captive Portal Authentication” on page 334

􀁺 VPN authentication: “VPN Configuration” on page 357

􀁺 MAC authentication: “Configuring MAC-Based Authentication” on page 393

􀁺 Stateful 802.1x, stateful NTLM, and WISPr authentication: “Stateful and WISPr Authentication” on

page 315

Overview of 802.1x Authentication

802.1x authentication consists of three components:

􀁺 The *supplicant*, or client, is the device attempting to gain access to the network. You can configure the

Aruba user-centric network to support 802.1x authentication for wired users as well as wireless users.

􀁺 The *authenticator* is the gatekeeper to the network and permits or denies access to the supplicants.

􀁺 The *Aruba controller* acts as the authenticator, relaying information between the authentication server

and supplicant. The EAP type must be consistent between the authentication server and supplicant and

is transparent to the controller.

The authentication server provides a database of information required for authentication and informs

the authenticator to deny or permit access to the supplicant.

The 802.1x authentication server is typically an EAP-compliant Remote Access Dial-In User Service

(RADIUS) server which can authenticate either users (through passwords or certificates) or the client

computer.

An example of an 802.1x authentication server is the Internet Authentication Service (IAS) in Windows

(see http://technet.microsoft.com/en-us/library/cc759077(WS.10).aspx).

Aruba user-centric networks, you can terminate the 802.1x authentication on the controller. The

controller passes user authentication to its internal database or to a “backend” non-802.1x server. This

feature, also called “*AAA FastConnect*,” is useful for deployments where an 802.1x EAP-compliant

RADIUS server is not available or required for authentication.

268 | 802.1x Authentication ArubaOS 5.0 | User Guide

Supported EAP Types

The following is the list of supported EAP types.

􀁺 PEAP—Protected EAP (PEAP) is an 802.1x authentication method that uses server-side public key

certificates to authenticate clients with server. The PEAP authentication creates an encrypted SSL / TLS

tunnel between the client and the authentication server. The exchange of information is encrypted and

stored in the tunnel ensuring the user credentials are kept secure.

􀁺 EAP-GTC—The EAP-GTC (Generic Token Card) type uses clear text method to exchange authentication

controls between client and server. Since the authentication mechanism uses the one-time tokens

(generated by the card), this method of credential exchange is considered safe. In addition, EAP-GTC is

used in PEAP or TTLS tunnels in wireless environments. The EAP-GTC is described in RFC 2284.

􀁺 EAP-AKA—The EAP-AKA (Authentication and Key Agreement) authentication mechanism is typically

used in mobile networks that include Universal Mobile Telecommunication Systems (UMTS) and CDMA

2000. This method uses the information stored in the Subscriber Identity Module (SIM) for

authentication. The EAP-AKA is described in RFC 4187.

􀁺 EAP-FAST—The EAP-FAST (Flexible Authentication via Secure Tunneling) is an alternative

authentication method to PEAP. This method uses the Protected Access Credential (PAC) for verifying

clients on the network. The EAP-FAST is described in RFC 4851.

􀁺 EAP-MD5—The EAP-MD5 method verifies MD5 hash of a user password for authentication. This

method is commonly used in a trusted network. The EAP-MD5 is described in RFC 2284.

􀁺 EAP-SIM—The EAP-SIM (Subscriber Identity Module) uses Global System for Mobile Communication

(GSM) Subscriber Identity Module (SIM) for authentication and session key distribution. This

authentication mechanism includes network authentication, user anonymity support, result indication,

and fast re-authentication procedure. Complete details about this authentication mechanism is

described in RFC 4186.

􀁺 EAP-TLS—The EAP-TLS (Transport Layer Security) uses Public key Infrastructure (PKI) to set up

authentication with a RADIUS server or any authentication server. This method requires the use of a

client-side certificate for communicating with the authentication server. The EAP-TLS is described in

RFC 5216.

􀁺 EAP-TLV- The EAP-TLV (type-length-value) method allows you to add additional information in an EAP

message. Often this method is used to provide more information about a EAP message. For example,

status information or authorization data. This method is always used after a typical EAP authentication

process.

􀁺 EAP-TTLS—The EAP-TTLS (Tunneled Transport Layer Security) method uses server-side certificates to

set up authentication between clients and servers. The actually authentication is, however, performed

using passwords. Complete details about EAP-TTLS is described in RFC 5281.

􀁺 LEAP—Lightweight Extensible Authentication Protocol (LEAP) uses dynamic WEP keys and mutual

authentication between client and RADIUS server.

􀁺 ZLXEAP—This is Zonelabs EAP. For more information, visit http://tools.ietf.org/html/draft-bersani-eapsynthesis-

sharedkeymethods-00#page-30.

Authentication with a RADIUS Server

See Table 51 for an overview of the parameters that you need to configure on authentication components

when the authentication server is an 802.1x EAP-compliant RADIUS server.

ArubaOS 5.0 | User Guide 802.1x Authentication | 269

Figure 43 *802.1x Authentication with RADIUS Server*

The supplicant and authentication server must be configured to use the same EAP type. The controller does

not need to know the EAP type used between the supplicant and authentication server.

For the controller to communicate with the authentication server, you must configure the IP address,

authentication port, and accounting port of the server on the controller. The authentication server must be

configured with the IP address of the RADIUS client, which is the controller in this case. Both the controller

and the authentication server must be configured to use the same shared secret.

The client communicates with the controller through a GRE tunnel in order to form an association with an

AP and to authenticate to the network. Therefore, the network authentication and encryption configured

for an ESSID must be the same on both the client and the controller.

Authentication Terminated on Controller

User authentication is performed either via the controller’s internal database or a non-802.1x server. See

“802.1x Authentication Profile Basic WebUI Parameters” on page 271 for an overview of the parameters that

you need to configure on 802.1x authentication components when 802.1x authentication is terminated on

the controller (AAA FastConnect).

Figure 44 *802.1x Authentication with Termination on Controller*

Client

(Supplicant)

WLAN Switch

(Authenticator)

Authentication

Server

• Server IP

• Shared Secret

• Auth Port

• Acct Port

• Client IP

• Shared Secret

• ESSID

• Network Authentication

• Data Encryption

• ESSID

• Network Authentication

• Data Encryption

• EAP Type • EAP Type

N O T E

Additional information on EAP types supported in a Windows environment, Microsoft supplicants, and

authentication server, is available at http://technet.microsoft.com/en-us/library/cc782851(WS.10).aspx.

Client

(Supplicant)

WLAN Switch

(Authenticator and

Authentication Server)

• EAP Type = EAP-TLS

or EAP-PEAP

• ESSID

• Network Authentication

• Data Encryption

User authentication via

internal database or non-

802.1x server

• EAP Type = EAP-TLS

or EAP-PEAP

• ESSID

• Network Authentication

• Data Encryption

270 | 802.1x Authentication ArubaOS 5.0 | User Guide

In this scenario, the supplicant is configured for EAP-Transport Layer Security (TLS) or EAP-Protected EAP

(PEAP).

􀁺 EAP-TLS is used with smart card user authentication. A smart card holds a digital certificate which, with

the user-entered personal identification number (PIN), allows the user to be authenticated on the

network. EAP-TLS relies on digital certificates to verify the identities of both the client and server.

EAP-TLS requires that you import server and certification authority (CA) certificates onto the controller

(see “Using Certificates with AAA FastConnect” on page 276). The client certificate is verified on the

controller (the client certificate must be signed by a known CA) before the user name is checked on the

authentication server.

􀁺 EAP-PEAP uses TLS to create an encrypted tunnel. Within the tunnel, one of the following “inner EAP”

methods is used:

􀂄 EAP-Generic Token Card (GTC): Described in RFC 2284, this EAP method permits the transfer of

unencrypted usernames and passwords from client to server. The main uses for EAP-GTC are onetime

token cards such as SecureID and the use of an LDAP or RADIUS server as the user

authentication server. You can also enable caching of user credentials on the controller as a backup

to an external authentication server.

􀂄 EAP-Microsoft Challenge Handshake Authentication Protocol version 2 (MS-CHAPv2): Described in

RFC 2759, this EAP method is widely supported by Microsoft clients. A RADIUS server must be used

as the backend authentication server.

If you are using the controller’s internal database for user authentication, you need to add the names and

passwords of the users to be authenticated. If you are using an LDAP server for user authentication, you

need to configure the LDAP server on the controller, and configure user IDs and passwords. If you are using

a RADIUS server for user authentication, you need to configure the RADIUS server on the controller.

Configuring 802.1x Authentication

On the controller, use the following steps to configure a wireless network that uses 802.1x authentication:

1. Configure the VLANs to which the authenticated users will be assigned. See Chapter 3, “Network

Parameters”

2. Configure policies and roles. You can specify a default role for users who are successfully authenticated

using 802.1x. You can also configure server derivation rules to assign a user role based on attributes

returned by the authentication server; server-derived user roles take precedence over default roles. For

more information about policies and roles, see Chapter 11, “Roles and Policies”.

3. Configure the authentication server(s) and server group. The server can be an 802.1x RADIUS server or,

if you are using AAA FastConnect, a non-802.1x server or the controller’s internal database. If you are

using EAP-GTC within a PEAP tunnel, you can configure an LDAP or RADIUS server as the

authentication server (see Chapter 9, “Authentication Servers”) If you are using EAP-TLS, you need to

import server and CA certificates on the controller (see “Using Certificates with AAA FastConnect” on

page 276).

4. Configure the AAA profile.

􀂄 Select the 802.1x default user role.

􀂄 Select the server group you previously configured for the 802.1x authentication server group.

N O T E

The Policy Enforcement Firewall Virtual Private Network (PEFV) module provides identity-based security for wired and

wireless users and must be installed on the controller. The stateful firewall allows user classification based on user

identity, device type, location and time of day and provides differentiated access for different classes of users. For

information about obtaining and installing licenses, see Chapter 28, “Software Licenses”.

ArubaOS 5.0 | User Guide 802.1x Authentication | 271

5. Configure the 802.1x authentication profile. See “Using the WebUI to configure 802.1x authentication”

on page 293

6. Configure the virtual AP profile for an AP group or for a specific AP:

􀂄 Select the AAA profile you previously configured.

􀂄 In the SSID profile, configure the WLAN for 802.1x authentication.

For details on how to complete the above steps, see “Example Configurations” on page 278

Using the WebUI to configure an 802.1x authentication profile

This section describes how to create and configure a new instance of an 802.1x authentication profile in the

WebUI or the CLI.

1. Navigate to the **Configuration > Security > Authentication > L2 Authentication** page.

2. In the Profiles list, select 802.1x Authentication Profile.

3. Enter a name for the profile, then click **Add**.

4. Click **Apply**.

5. In the Profiles list, select the 802.1x authentication profile you just created.

6. The profile details window includes **Basic** and **Advanced** tabs for basic and advanced configuration

settings. Click on one or both of these tab to configure the 802.1x Authentication settings. Table 51

describes the parameters you can configure in the high-throughput radio profile.

Table 51 *802.1x Authentication Profile Basic WebUI Parameters*

Parameter Description

Basic 802.1x Authentication Profile settings

Max authentication

failures

Number of times a user can try to login with wrong credentials after which the user will

be blacklisted as a security threat.

Set to 0 to disable blacklisting, otherwise enter a non-zero integer to blacklist the user

after the specified number of failures.

Default: 0

Enforce Machine

Authentication

(For Windows environments only) Select this option to enforce machine authentication

before user authentication. If selected, either the Machine Authentication Default Role

or the User Authentication Default Role is assigned to the user, depending on which

authentication is successful. This option is disabled by default.

Note: This option may require a license (see Chapter 28 on page 551).The Enforce

Machine Authentication checkbox is also available on the Advanced settings tab.

Machine Authentication:

Default Machine Role

Select the default role to be assigned to the user after completing only machine

authentication.

Default: guest

Machine Authentication:

Default User Role

Select the default role to be assigned to the user after completing 802.1x

authentication.

Default: guest

Reauthentication Select this option to force the client to do a 802.1x re-authentication after the

expiration of the default timer for re-authentication. The default value of the timer

(Reauthentication Interval) is 24 hours. If the user fails to re-authenticate with valid

credentials, the state of the user is cleared.

If derivation rules are used to classify 802.1x-authenticated users, then the Reauthentication

timer per role overrides this setting.

Default: disabled

272 | 802.1x Authentication ArubaOS 5.0 | User Guide

Termination Select this option to terminate 802.1x authentication on the controller.

Default: disabled

Termination EAP-Type The EAP method, either EAP-PEAP or EAP-TLS.

Default: eap-peap

Termination Inner EAPType

Select one of the following:

􀁺 EAP-Generic Token Card (GTC): Described in RFC 2284, this EAP method permits

the transfer of unencrypted usernames and passwords from client to server. The

main uses for EAP-GTC are one-time token cards such as SecureID and the use of

LDAP or RADIUS as the user authentication server. You can also enable caching of

user credentials on the controller as a backup to an external authentication server.

􀁺 EAP-Microsoft Challenge Authentication Protocol version 2 (MS-CHAPv2):

Described in RFC 2759, this EAP method is widely supported by Microsoft clients.

Default: eap-mschapv2

Advanced 802.1x Authentication Profile settings

Max authentication

failures

Number of times a user can try to login with wrong credentials

after which the user is blacklisted as a security threat. Set to 0

to disable blacklisting, otherwise enter a non-zero integer to

blacklist the user after the specified number of failures. The range of allowed values is

0-5 failures, and the default value is 0 failures.

Note: This option may require a license (see Chapter 28 on page 551).

Enforce Machine

Authentication

Select the Enforce Machine Authentication option to require

machine authentication. This option is also available on the Basic settings tab.

Note: This option may require a license (see Chapter 28 on page 551).

Machine Authentication:

Default Machine Role

Default role assigned to the user after completing only machine authentication. The

default role for this setting is the “guest” role.

Machine Authentication

Cache Timeout

The timeout, in hours, for machine authentication. The allowed range of values is 1-

1000 hours, and the default value is 24 hours.

Blacklist on Machine

Authentication Failure

Select the Blacklist on Machine Authentication Failure checkbox to blacklist a client

if machine authentication fails. This setting is disabled by default

Machine Authentication:

Default User Role

Default role assigned to the user after 802.1x authentication. The default role for this

setting is the “guest” role.

Interval between Identity

Requests

Interval, in seconds, between identity request retries. The allowed range of values is 1-

65535 seconds, and the default value is 30 seconds.

Quiet Period after Failed

Authentication

The enforced quiet period interval, in seconds, following failed authentication. The

allowed range of values is 1-65535 seconds, and the default value is 30 seconds.

Reauthentication Interval Interval, in seconds, between reauthentication attempts. The allowed range of values

for this parameter is 60-864000 seconds, and the default value is 86400 seconds

(1day).

Use Server provided

Reauthentication Interval

Select this option to override any user-defined reauthentication interval and use the

reauthentication period defined by the authentication server.

Multicast Key Rotation

Time Interval

Interval, in seconds, between multicast key rotation. The allowed range of values for

this parameter is 60-864000 seconds, and the default value is 1800 seconds.

Table 51 *802.1x Authentication Profile Basic WebUI Parameters (Continued)*

Parameter Description

ArubaOS 5.0 | User Guide 802.1x Authentication | 273

Unicast Key Rotation

Time Interval

Interval, in seconds, between unicast key rotation. The allowed range of values for this

parameter is 60-864000 seconds, and the default value is 900 seconds.

Authentication Server

Retry Interval

Server group retry interval, in seconds. The allowed range of values for this parameter

is 5-65535 seconds, and the default value is 30 seconds.

Authentication Server

Retry Count

Maximum number of authentication requests that are sent to server group.

The allowed range of values for this parameter is 0-3 requests, and the default value is

2 requests.

Framed MTU Sets the framed Maximum Transmission Unit (MTU) attribute sent to the authentication

server.

The allowed range of values for this parameter is 500-1500 bytes, and the default value

is 1100 bytes.

Number of times IDRequests

are retried

Maximum number of times ID requests are sent to the client. The allowed range of

values for this parameter is 1-10 retries, and the default value is 3 retries.

Maximum Number of

Reauthentication

Attempts

Number of times a user can try to login with wrong credentials after which the user is

blacklisted as a security threat. Set to 0 to disable blacklisting, otherwise enter a value

from 0-5 to blacklist the user after the specified number of failures.

Note: If changed from its default value, this may require a license (see Chapter 28 on

page 551).

Maximum number of

times Held State can be

bypassed

Number of consecutive authentication failures which, when reached, causes the

controller to not respond to authentication requests from a client while the controller is

in a held state after the authentication failure. Before this number is reached, the

controller responds to authentication requests from the client even while the controller

is in its held state.

(This parameter is applicable when 802.1x authentication is terminated on the

controller, also known as AAA FastConnect.)

The allowed range of values for this parameter is 0-3 failures, and the default value is 0.

Dynamic WEP Key

Message Retry Count

Set the Number of times WPA/WPA2 Key Messages are retried. The allowed range of

values is 1-5 retries, and the default value is 3 retries.

Dynamic WEP Key Size The default dynamic WEP key size is 128 bits, If desired, you can change this

parameter to either 40 bits.

Interval between WPA/

WPA2 Key Messages

Interval, in milliseconds, between each WPA key exchange.s The allowed range of

values is 1000-5000ms, and the default value is 3000 ms.

Delay between EAPSuccess

and WPA2

Unicast Key Exchange

Interval, in milliseconds, between unicast and multicast key exchanges. The allowed

range of values is 0-2000ms, and the default value is 0 ms (no delay).

Delay between WPA/

WPA2 Unicast Key and

Group Key Exchange

Interval, in milliseconds, between unicast and multicast key exchanges. The allowed

range of values is 0-2000ms, and the default value is 0 ms (no delay).

WPA/WPA2 Key Message

Retry Count

Number of times WPA/WPA2 key messages are retried. The allowed range of values for

this parameter is 1-5 retries, and the default value is 3 retries.

Multicast Key Rotation Select this checkbox to enable multicast key rotation. This feature is disabled by

default.

Unicast Key Rotation Select this checkbox to enable unicast key rotation. This feature is disabled by default.

Table 51 *802.1x Authentication Profile Basic WebUI Parameters (Continued)*

Parameter Description

274 | 802.1x Authentication ArubaOS 5.0 | User Guide

Reauthentication Select the Reauthentication checkbox to force the client to do a 802.1x

reauthentication after the expiration of the default timer for reauthentication. (The

default value of the timer is 24 hours.) If the user fails to reauthenticate with valid

credentials, the state of the user is cleared. If derivation rules are used to classify

802.1x-authenticated users, then the reauthentication timer per role overrides this

setting.

This option is disabled by default.

Opportunistic Key

Caching

By default, the 802.1x authentication profile enables a cached pairwise master key

(PMK) derived via a client and an associated AP and used when the client roams to a

new AP. This allows clients faster roaming without a full 802.1x authentication.

Uncheck this option to disable this feature.

Note: Make sure that the wireless client (the 802.1x supplicant) supports this feature. If

the client does not support this feature, the client will attempt to renegotiate the key

whenever it roams to a new AP. As a result, the key cached on the controller can be out

of sync with the key used by the client.

Validate PMKID If opp-key-caching is enabled, this option instructs the controller to check the pairwise

master key (PMK) ID sent by the client. When this option is enabled, the client must

send a PMKID in the associate or reassociate frame to indicate that it supports OKC;

otherwise, full 802.1x authentication takes place. (This feature is optional and is

disabled by default, since most clients that support OKC do not send the PMKID in

their association request.)

Use Session Key Select the Use Session Key option to use the RADIUS session key as the unicast

WEP key. This option is disabled by default.

Use Static Key Select the Use Static Key option to use a static key as the unicast/multicast WEP key.

This option is disabled by default.

xSec MTU Set the maximum transmission unit (MTU) for frames using the xSec protocol. The

range of allowed values is 1024-1500 bytes, and 1300 bytes

Termination Select the Termination checkbox to allow 802.1x authentication to terminate on the

controller. This option is disabled by default.

Termination EAP-Type If termination is enabled, click either EAP-PEAP or EAP-TLS to select a Extensible

Authentication Protocol (EAP) method.

Termination Inner EAPType

If you are using EAP-PEAP as the EAP method, specify one of the following

inner EAP types:

􀁺 eap-gtc: Described in RFC 2284, this EAP method permits the transfer of

unencrypted

􀁺 usernames and passwords from client to server. The main uses for EAP-GTC are

one-time token cards such as SecureID and the use of LDAP or RADIUS as the

user authentication server. You can also enable caching of user credentials on the

controller as a backup to an external authentication server.

􀁺 eap-mschapv2: Described in RFC 2759, this EAP method is widely supported by

Microsoft clients.

Token Caching If you select EAP-GTC as the inner EAP method, you can select the Token Caching

checkbox to enable the controller to cache the username and password of each

authenticated user. The controller continues to reauthenticate users with the remote

authentication server, however, if the authentication server is not available, the

controller will inspect its cached credentials to reauthenticate users.

This option is disabled by default.

Token Caching Period If you select EAP-GTC as the inner EAP method, you can specify the timeout period, in

hours, for the cached information. The default value is 24 hours.

Table 51 *802.1x Authentication Profile Basic WebUI Parameters (Continued)*

Parameter Description

ArubaOS 5.0 | User Guide 802.1x Authentication | 275

7. Click **Apply**.

Using the CLI to configure an 802.1x authentication profile

The following command configures settings for an 802.1x authentication profiles. Individual parameters are

described in Table 51, above.

aaa authentication dot1x {<profile>|countermeasures}

ca-cert <certificate>

clear

clone <profile>

eapol-logoff

framed-mtu <mtu>

heldstate-bypass-counter <number>

ignore-eap-id-match

ignore-eapolstart-afterauthentication

machine-authentication blacklist-on-failure|{cache-timeout <hours>}|enable|

{machine-default-role <role>}|{user-default-role <role>}

max-authentication-failures <number>

max-requests <number>

multicast-keyrotation

no ...

opp-key-caching

reauth-max <number>

reauthentication

server {server-retry <number>|server-retry-period <seconds>}

server-cert <certificate>

CA-Certificate Click the CA-Certificate drop-down list and select a certificate for client

authentication. The CA certificate needs to be loaded in the controller before it will

appear on this list.

Server-Certificate Click the Server-Certificate drop-down list and select a server certificate the

controller will use to authenticate itself to the client.

TLS Guest Access Select TLS Guest Access to enable guest access for EAP-TLS users with valid

certificates. This option is disabled by default.

TLS Guest Role Click the TLS Guest Role drop-down list and select the default user role for EAP-TLS

guest users.

Note: This option may require a license (see Chapter 28 on page 551)..

Ignore EAPOL-START

after authentication

Select Ignore EAPOL-START after authentication to ignore EAPOL-START messages

after authentication. This option is disabled by default.

Handle EAPOL-Logoff Select Handle EAPOL-Logoff to enable handling of EAPOL-LOGOFF messages. This

option is disabled by default.

Ignore EAP ID during

negotiation

Select Ignore EAP ID during negotiation to ignore EAP IDs during negotiation. This

option is disabled by default.

WPA-Fast-Handover Select this option to enable WPA-fast-handover on phones that support this feature.

WAP fast-handover is disabled by default.

Disable rekey and

reauthentication for

clients on call

This feature disables rekey and reauthentication for VoWLAN clients. It is disabled by

default, meaning that rekey and reauthentication is enabled.

Note: This option may require a license (see Chapter 28 on page 551).

Table 51 *802.1x Authentication Profile Basic WebUI Parameters (Continued)*

Parameter Description

276 | 802.1x Authentication ArubaOS 5.0 | User Guide

termination {eap-type <type>}|enable|enable-token-caching|{inner-eap-type (eapgtc|

eap-mschapv2)}|{token-caching-period <hours>}

timer {idrequest\_period <seconds>}|{mkey-rotation-period <seconds>}|{quiet-period

<seconds>}|{reauth-period <seconds>}|{ukey-rotation-period <seconds>}|{wpagroupkeydelay

<seconds>}|{wpa-key-period <milliseconds>}

tls-guest-access

tls-guest-role <role>

unicast-keyrotation

use-session-key

use-static-key

validate-pmkid

voice-aware

wep-key-retries <number>

wep-key-size {40|128}

wpa-fast-handover

wpa-key-retries <number>

xSec-mtu <mtu>

Using Certificates with AAA FastConnect

The controller supports 802.1x authentication using digital certificates for AAA FastConnect.

􀁺 Server Certificate—A server certificate installed in the controller verifies the authenticity of the

controller for 802.1x authentication. Aruba controllers ship with a demonstration digital certificate.

Until you install a customer-specific server certificate in the controller, this demonstration certificate is

used by default for all secure HTTP connections (such as the WebUI and captive portal) and AAA

FastConnect. This certificate is included primarily for the purposes of feature demonstration and

convenience and is not intended for long-term use in production networks. Users in a production

environment are urged to obtain and install a certificate issued for their site or domain by a well-known

certificate authority (CA). You can generate a Certificate Signing Request (CSR) on the controller to

submit to a CA. For information on how to generate a CSR and how to import the CA-signed certificate

into the controller, see “Managing Certificates” on page 526

􀁺 Client Certificates—Client certificates are verified on the controller (the client certificate must be signed

by a known CA) before the user name is checked on the authentication server. To use client certificate

authentication for AAA FastConnect, you need to import the following certificates into the controller

(see “Importing Certificates” on page 528):

􀂄 Controller’s server certificate

􀂄 CA certificate for the CA that signed the client certificates

Using the WebUI to configure AAA FastConnect certificate authentication:

1. Navigate to the **Configuration > Security > Authentication > L2 Authentication** page.

2. In the Profiles list, select **802.1x Authentication Profile**.

3. Select the “default” 802.1x authentication profile from the drop-down menu to display configuration

parameters.

4. In the **Basic** tab, select **Termination**.

5. Select the **Advanced** Tab.

6. In the Server-Certificate field, select the server certificate imported into the controller.

7. In the CA-Certificate field, select the CA certificate imported into the controller.

8. Click **Save As**. Enter a name for the 802.1x authentication profile.

9. Click **Apply**.

ArubaOS 5.0 | User Guide 802.1x Authentication | 277

Using the CLI to configure AAA FastConnect certificate authentication:

aaa authentication dot1x <profile>

termination enable

server-cert <certificate>

ca-cert <certificate>

Configuring User and Machine Authentication

When a Windows device boots, it logs onto the network domain using a machine account. Within the

domain, the device is authenticated before computer group policies and software settings can be executed;

this process is known as *machine authentication*. Machine authentication ensures that only authorized

devices are allowed on the network.

You can configure 802.1x for both user and machine authentication (select the **Enforce Machine**

**Authentication** option described in Table 51 on page 271). This tightens the authentication process further

since both the device and user need to be authenticated.

Role Assignment with Machine Authentication Enabled

When you enable machine authentication, there are two additional roles you can define in the 802.1x

authentication profile:

􀁺 Machine authentication default machine role

􀁺 Machine authentication default user role

While you can select the same role for both options, you should define the roles as per the polices that need

to be enforced. Also, these roles can be different from the 802.1x authentication default role configured in

the AAA profile.

With machine authentication enabled, the assigned role depends upon the success or failure of the machine

and user authentications. In certain cases, the role that is ultimately assigned to a client can also depend

upon attributes returned by the authentication server or server derivation rules configured on the

controller.

Table 52 describes role assignment based on the results of the machine and user authentications.

Table 52 *Role Assignment for User and Machine Authentication*

Machine

Auth

Status

User

Auth

Status

Description Role Assigned

Failed Failed Both machine authentication and user

authentication failed. L2 authentication failed.

No role assigned. No access to the

network allowed.

Failed Passed Machine authentication fails (for example, the

machine information is not present on the server)

and user authentication succeeds. Server-derived

roles do not apply.

Machine authentication default user

role configured in the 802.1x

authentication profile.

Passed Failed Machine authentication succeeds and user

authentication has not been initiated. Serverderived

roles do not apply.

Machine authentication default

machine role configured in the 802.1x

authentication profile.

Passed Passed Both machine and user are successfully

authenticated. If there are server-derived roles,

the role assigned via the derivation take

precedence. This is the *only* case where serverderived

roles are applied.

A role derived from the authentication

server takes precedence. Otherwise,

the 802.1x authentication default role

configured in the AAA profile is

assigned.

278 | 802.1x Authentication ArubaOS 5.0 | User Guide

For example, if the following roles are configured:

􀁺 802.1x authentication default role (in AAA profile): dot1x\_user

􀁺 Machine authentication default machine role (in 802.1x authentication profile): dot1x\_mc

􀁺 Machine authentication default user role (in 802.1x authentication profile): guest

Role assignments would be as follows:

􀁺 If both machine and user authentication succeed, the role is dot1x\_user. If there is a server-derived role,

the server-derived role takes precedence.

􀁺 If only machine authentication succeeds, the role is dot1x\_mc.

􀁺 If only user authentication succeeds, the role is guest.

􀁺 On failure of both machine and user authentication, the user does not have access to the network.

VLAN Assignment with Machine Authentication Enabled

With machine authentication enabled, the VLAN to which a client is assigned (and from which the client

obtains its IP address) depends upon the success or failure of the machine and user authentications. The

VLAN that is ultimately assigned to a client can also depend upon attributes returned by the authentication

server or server derivation rules configured on the controller (see “About VLAN Assignments” on page 62).

If machine authentication is successful, the client is assigned the VLAN configured in the virtual AP profile.

However, the client can be assigned a derived VLAN upon successful user authentication.

Table 53 describes VLAN assignment based on the results of the machine and user authentications when

VLAN derivation is used.

Example Configurations

The following examples show basic configurations on the controller for:

􀁺 “Authentication with an 802.1x RADIUS Server” on page 319

􀁺 “Authentication with the Controller’s Internal Database” on page 333

N O T E

You can optionally assign a VLAN as part of a user role configuration. You should not use VLAN derivation if you

configure user roles with VLAN assignments

Table 53 *VLAN Assignment for User and Machine Authentication*

Machine Auth

Status

User Auth

Status

Description VLAN Assigned

Failed Failed Both machine authentication and user

authentication failed. L2 authentication failed.

No VLAN

Failed Passed Machine authentication fails (for example, the

machine information is not present on the

server) and user authentication succeeds.

VLAN configured in the

virtual AP profile

Passed Failed Machine authentication succeeds and user

authentication has not been initiated.

VLAN configured in the

virtual AP profile

Passed Passed Both machine and user are successfully

authenticated.

Derived VLAN.

Otherwise, VLAN

configured in the virtual

AP profile.

ArubaOS 5.0 | User Guide 802.1x Authentication | 279

In the following examples:

􀁺 Wireless clients associate to the ESSID **WLAN-01**.

􀁺 The following roles allow different networks access capabilities:

􀂄 student

􀂄 faculty

􀂄 guest

􀂄 system administrators

The examples show how to configure using the WebUI and CLI commands.

Authentication with an 802.1x RADIUS Server

􀁺 An EAP-compliant RADIUS server provides the 802.1x authentication. The RADIUS server administrator

must configure the server to support this authentication. The administrator must also configure the

server to all communications with the Aruba controller.

􀁺 The authentication type is WPA. From the 802.1x authentication exchange, the client and the controller

derive dynamic keys to encrypt data transmitted on the wireless network.

􀁺 802.1x authentication based on PEAP with MS-CHAPv2 provides both computer and user

authentication. If a user attempts to log in without the computer being authenticated first, the user is

placed into a more limited “guest” user role.

Windows domain credentials are used for computer authentication, and the user’s Windows login and

password are used for user authentication. A single user sign-on facilitates both authentication to the

wireless network and access to the Windows server resources.

Configuring Policies and Roles

Create the following policies and user roles:

􀁺 The **student** policy prevents students from using telnet, POP3, FTP, SMTP, SNMP, or SSH to the wired

portion of the network. The **student** policy is mapped to the **student** user role.

􀁺 The **faculty** policy is similar to the **student** policy, however faculty members are allowed to use POP3

and SMTP for VPN remote access from home. (Students are not permitted to use VPN remote access.)

The **faculty** policy is mapped to the **faculty** user role.

􀁺 The **guest** policy permits only access to the Internet (via HTTP or HTTPS) and only during daytime

working hours. The **guest** policy is mapped to the **guest** user role.

􀁺 The **allowall** policy, a predefined policy, allows unrestricted access to the network. The **allowall** policy

is mapped to both the **sysadmin** user role and the **computer** user role.

Using the Web to create the student policy and role

1. Navigate to the **Configuration > Security > Access Control > Policies** page. Select **Add** to add the

student policy.

2. For Policy Name, enter **student**.

3. For Policy Type, select **IPv4 Session**.

4. Under Rules, select **Add** to add rules for the policy.

a. Under Source, select **user**.

N O T E

Appendix D, “802.1x Configuration for IAS and Windows Clients”describes how to configure the Microsoft Internet

Authentication Server and Windows XP wireless client to operate with the controller configuration shown in this

section.

280 | 802.1x Authentication ArubaOS 5.0 | User Guide

b. Under Destination, select **alias**.

c. Under the alias selection, click **New**. For Destination Name, enter “Internal Network”. Click **Add** to

add a rule. For Rule Type, select **network**. For IP Address, enter 10.0.0.0. For Network Mask/Range,

enter 255.0.0.0. Click **Add** to add the network range. Repeat these steps to add the network range

172.16.0.0 255.255.0.0. Click **Done**. The alias “Internal Network” appears in the Destination menu.

This step defines an alias representing all internal network addresses. Once defined, you can use the

alias for other rules and policies.

d. Under Destination, select Internal Network.

e. Under Service, select **service**. In the Service scrolling list, select **svc-telnet**.

f. Under Action, select **drop**.

g. Click **Add**.

5. Under Rules, click **Add**.

a. Under Source, select **user**.

b. Under Destination, select **alias**. Then select Internal Network.

c. Under Service, select **service**. In the Service scrolling list, select **svc-pop3**.

d. Under Action, select **drop**.

e. Click **Add**.

6. Repeat steps 4A-E to create rules for the following services: svc-ftp, svc-smtp, svc-snmp, and svc-ssh.

7. Click **Apply**.

8. Click the **User Roles** tab. Click **Add** to create the student role.

9. For Role Name, enter **student**.

10. Under Firewall Policies, click **Add**. In Choose from Configured Policies, select the student policy you

previously created. Click **Done**.

11. Click **Apply**.

Using the WebUI to create the faculty policy and role

1. Navigate to the **Configuration > Security > Access Control > Policies** page. Click **Add** to add the

faculty policy.

2. For Policy Name, enter **faculty**.

3. For Policy Type, select **IPv4 Session**.

4. Under Rules, click **Add** to add rules for the policy.

a. Under Source, select **user**.

b. Under Destination, select alias, then select **Internal Network**.

c. Under Service, select **service**. In the Service scrolling list, select **svc-telnet**.

d. Under Action, select **drop**.

e. Click **Add**.

f. Repeat steps A-E to create rules for the following services: svc-ftp, svc-snmp, and svc-ssh.

5. Click **Apply**.

6. Select the **User Roles** tab. Click **Add** to create the faculty role.

7. For Role Name, enter **faculty**.

8. Under **Firewall Policies**, click **Add**. In Choose from Configured Policies, select the faculty policy you

previously created. Click **Done**.

ArubaOS 5.0 | User Guide 802.1x Authentication | 281

Using the WebUI to create the guest policy and role

1. Navigate to the **Configuration > Security > Access Control > Time Ranges** page to define the time

range “working-hours”. Click **Add**.

a. For Name, enter **working-hours**.

b. For Type, select **Periodic**.

c. Click **Add**.

d. For Start Day, click **Weekday**.

e. For Start Time, enter **07:30**.

f. For End Time, enter **17:00**.

g. Click **Done**.

h. Click **Apply**.

2. Click the **Policies** tab. Click **Add** to add the guest policy.

3. For Policy Name, enter **guest**.

4. For Policy Type, select **IPv4 Session**.

5. Under Rules, click **Add** to add rules for the policy.

To create rules to permit access to DHCP and DNS servers during working hours:

a. Under Source, select **user**.

b. Under Destination, select **host**. In Host IP, enter **10.1.1.25**.

c. Under Service, select **service**. In the Service scrolling list, select **svc-dhcp**.

d. Under Action, select **permit**.

e. Under Time Range, select **working-hours**.

f. Click **Add**.

g. Repeat steps A-F to create a rule for svc-dns.

To create a rule to deny access to the internal network:

a. Under Source, select **user**.

b. Under Destination, select **alias**. Select **Internal Network**.

c. Under Service, select **any**.

d. Under Action, select **drop**.

e. Click **Add**.

To create rules to permit HTTP and HTTPS access during working hours:

a. Under Source, select **user**.

b. Under Destination, select **any**.

c. Under Service, select service. In the Services scrolling list, select **svc-http**.

d. Under Action, select **permit**.

e. Under Time Range, select **working-hours**.

f. Click **Add**.

g. Repeat steps A-F for the svc-https service.

To create a rule that denies the user access to all destinations and all services:

a. Under Source, select **user**.

b. Under Destination, select **any**.

c. Under Service, select **any**.

282 | 802.1x Authentication ArubaOS 5.0 | User Guide

d. Under Action, select **drop**.

e. Click **Add**.

6. Click **Apply**.

7. Click the **User Roles** tab. Click **Add** to create the guest role.

8. For Role Name, enter **guest**.

9. Under **Firewall Policies**, click **Add**. In Choose from Configured Policies, select the guest policy you

previously created. Click **Done**.

Using the WebUI to create the sysadmin role

1. Navigate to **Configuration > Security > Access Control > User Roles** page. Click **Add** to create the

sysadmin role.

2. For Role Name, enter **sysadmin**.

3. Under Firewall Policies, click **Add**. In Choose from Configured Policies, select the predefined **allowall**

policy. Click **Done**.

4. Click **Apply**.

Using the WebUI to create the computer role

1. Navigate to **Configuration > Security > Access Control > User Roles** page. Click **Add** to create the

computer role.

2. For Role Name, enter **computer**.

3. Under Firewall Policies, click **Add**. In Choose from Configured Policies, select the predefined **allowall**

policy. Click **Done**.

4. Click **Apply**.

Using the CLI to create an alias for the internal network

netdestination “Internal Network”

network 10.0.0.0 255.0.0.0

network 172.16.0.0 255.255.0.0

Using the CLI to create the student role

ip access-list session student

user alias “Internal Network” svc-telnet deny

user alias “Internal Network” svc-pop3 deny

user alias “Internal Network” svc-ftp deny

user alias “Internal Network” svc-smtp deny

user alias “Internal Network” svc-snmp deny

user alias “Internal Network” svc-ssh deny

user-role student

session-acl student

session-acl allowall

Using the CLI to create the faculty role

ip access-list session faculty

user alias “Internal Network” svc-telnet deny

user alias “Internal Network” svc-ftp deny

user alias “Internal Network” svc-snmp deny

user alias “Internal Network” svc-ssh deny

user-role faculty

session-acl faculty

session-acl allowall

ArubaOS 5.0 | User Guide 802.1x Authentication | 283

Using the CLI to create the guest role

time-range working-hours periodic

weekday 07:30 to 17:00

ip access-list session guest

user host 10.1.1.25 svc-dhcp permit time-range working-hours

user host 10.1.1.25 svc-dns permit time-range working-hours

user alias “Internal Network” any deny

user any svc-http permit time-range working-hours

user any svc-https permit time-range working-hours

user any any deny

user-role guest

session-acl guest

Using the CLI to create the sysadmin role

user-role sysadmin

session-acl allowall

Using the CLI to create the computer role

user-role computer

session-acl allowall

Configuring the RADIUS Authentication Server

Configure the RADIUS server IAS1, with IP address 10.1.1.21 and shared key. The RADIUS server is

configured to sent an attribute called Class to the controller; the value of this attribute is set to either

“student,” “faculty,” or “sysadmin” to identify the user’s group. The controller uses the literal value of this

attribute to determine the role name.

On the controller, you add the configured server (IAS1) into a server group. For the server group, you

configure the server rule that allows the Class attribute returned by the server to set the user role.

Using the WebUI to configure the RADIUS authentication server

1. Navigate to the **Configuration > Security > Authentication > Servers** page.

2. In the Servers list, select Radius Server. In the RADIUS Server Instance list, enter **IAS1** and click **Add**.

a. Select IAS1 to display configuration parameters for the RADIUS server.

b. For IP Address, enter **10.1.1.21**.

c. For Key, enter **|\*a^t%183923!**. (You must enter the key string twice.)

d. Click **Apply**.

3. In the Servers list, select Server Group. In the Server Group Instance list, enter **IAS** and click **Add**.

a. Select the server group IAS to display configuration parameters for the server group.

b. Under Servers, click **New**.

c. From the Server Name drop-down menu, select IAS1. Click **Add Server**.

4. Under Server Rules, click **New**.

a. For Condition, enter **Class**.

b. For Attribute, select **value-of** from the drop-down menu.

c. For Operand, select **set role**.

d. Click **Add**.

5. Click **Apply**.

284 | 802.1x Authentication ArubaOS 5.0 | User Guide

Using the CLI to configure the RADIUS authentication server

aaa authentication-server radius IAS1

host 10.1.1.21

key |\*a^t%183923!

aaa server-group IAS

auth-server IAS1

set role condition Class value-of

Configure 802.1x Authentication

An AAA profile specifies the 802.1x authentication profile and 802.1x server group to be used for

authenticating clients for a WLAN. The AAA profile also specifies the default user roles for 802.1x and MAC

authentication.

In the 802.1x authentication profile, configure enforcement of machine authentication before user

authentication. If a user attempts to log in without machine authentication taking place first, the user is

placed in the limited guest role.

Using the WebUI to configure 802.1x authentication

1. Navigate to the **Configuration > Security > Authentication > L2 Authentication** page.

2. Select 802.1x Authentication Profile.

a. In the list of instances, enter **dot1x**, then click **Add**.

b. Select the profile name you just added.

c. Select **Enforce Machine Authentication**.

d. For the Machine Authentication: Default Machine Role, select **computer**.

e. For the Machine Authentication: Default User Role, select **guest**.

f. Click **Apply**.

3. Select the **AAA Profiles** tab.

a. In the AAA Profiles Summary, click **Add** to add a new profile.

b. Enter **aaa\_dot1x**, then click **Add**.

a. Select the profile name you just added.

b. For MAC Auth Default Role, select **computer**.

c. For 802.1x Authentication Default Role, select **faculty**.

d. Click **Apply**.

4. In the Profiles list (under the aaa\_dot1x profile), select 802.1x Authentication Profile.

a. From the drop-down menu, select the **dot1x** 802.1x authentication profile you configured previously.

b. Click **Apply**.

5. In the Profiles list (under the aaa\_dot1x profile), select 802.1x Authentication Server Group.

a. From the drop-down menu, select the IAS server group you created previously.

b. Click **Apply**.

Using the CLI to configure 802.1x authentication

aaa authentication dot1x dot1x

machine-authentication enable

machine-authentication machine-default-role computer

machine-authentication user-default-role guest

aaa profile aaa\_dot1x

ArubaOS 5.0 | User Guide 802.1x Authentication | 285

dot1x-default-role faculty

mac-default-role computer

authentication-dot1x dot1x

dot1x-server-group IAS

Configure VLANs

In this example, wireless clients are assigned to either VLAN 60 or 61 while guest users are assigned to

VLAN 63. VLANs 60 and 61 split users into smaller IP subnetworks, improving performance by decreasing

broadcast traffic. The VLANs are internal to the Aruba controller only and do not extend into other parts of

the wired network. The clients’ default gateway is the Aruba controller, which routes traffic out to the

10.1.1.0 subnetwork.

You configure the VLANs, assign IP addresses to each VLAN, and establish the “helper address” to which

client DHCP requests are forwarded.

Using the WebUI to configure VLANs

1. Navigate to the **Configuration > Network > VLANs** page. Click **Add** to add VLAN 60.

a. For VLAN ID, enter **60**.

b. Click **Apply**.

c. Repeat steps A and B to add VLANs 61 and 63.

2. To configure IP parameters for the VLANs, navigate to the **Configuration > Network > IP > IP**

**Interfaces** page.

a. Click **Edit** for VLAN 60.

b. For IP Address, enter **10.1.60.1**.

c. For Net Mask, enter **255.255.255.0**.

d. Under DHCP Helper Address, click **Add**. Enter **10.1.1.25** and click **Add**.

e. Click **Apply**.

3. In the IP Interfaces page, click **Edit** for VLAN 61.

a. For IP Address, enter **10.1.61.1**.

b. For Net Mask, enter **255.255.255.0**.

c. Under DHCP Helper Address, click **Add**. Enter **10.1.1.25** and click **Add**.

d. Click **Apply**.

4. In the IP Interfaces page, click **Edit** for VLAN 63.

a. For IP Address, enter **10.1.63.1**.

b. For Net Mask, enter **255.255.255.0**.

c. Under DHCP Helper Address, click **Add**. Enter **10.1.1.25** and click **Add**.

d. Click **Apply**.

5. Select the **IP Routes** tab.

a. For Default Gateway, enter **10.1.1.254**.

b. Click **Apply**.

Using the CLI to Configure VLANs

vlan 60

interface vlan 60

ip address 10.1.60.1 255.255.255.0

ip helper-address 10.1.1.25

286 | 802.1x Authentication ArubaOS 5.0 | User Guide

vlan 61

interface vlan 61

ip address 10.1.61.1 255.255.255.0

ip helper-address 10.1.1.25

vlan 63

interface vlan 63

ip address 10.1.63.1 255.255.255.0

ip helper-address 10.1.1.25

ip default-gateway 10.1.1.254

Configure the WLANs

In this example, default AP parameters for the entire network are as follows: the default ESSID is WLAN-01

and the encryption mode is TKIP. A second ESSID called “guest” has the encryption mode set to static WEP

with a configured WEP key.

In this example, the non-guest clients that associate to an AP are mapped into one of two different user

VLANs. The initial AP to which the client associates determines the VLAN: clients that associate to APs in

the first floor of the building are mapped to VLAN 60 and clients that associate to APs in the second floor of

the building are mapped to VLAN 61. Therefore, the APs in the network are segregated into two AP groups,

named “first-floor” and “second-floor”. (See “AP Groups” on page 109 for information about creating AP

groups.) The guest clients are mapped into VLAN 63.

Guest WLAN

You create and configure the virtual AP profile “guest” and apply the profile to each AP group. The “guest”

virtual AP profile contains the SSID profile “guest” which configures static WEP with a WEP key.

Using the WebUI to configure the WLAN

1. Navigate to the **Configuration > Wireless > AP Configuration** page.

2. In the AP Group list, click **Edit** for first-floor.

3. Under Profiles, select Wireless LAN, then select Virtual AP.

4. To create the guest virtual AP:

a. Select NEW from the Add a profile drop-down menu. Enter **guest**, and click **Add**.

b. In the Profile Details entry for the guest virtual AP profile, select NEW from the SSID profile dropdown

menu. A pop-up window allows you to configure the SSID profile.

c. For the name for the SSID profile enter **guest**.

d. For the Network Name for the SSID, enter **guest**.

e. For Network Authentication, select **None**.

f. For Encryption, select **WEP**.

g. Enter the WEP Key.

h. Click **Apply** to apply the SSID profile to the Virtual AP.

i. Under Profile Details, click **Apply**.

5. Click on the **guest** virtual AP name in the Profiles list or in Profile Details to display configuration

parameters.

a. Make sure Virtual AP enable is selected.

b. For VLAN, select **63**.

c. Click **Apply**.

6. Navigate to the **Configuration > Wireless > AP Configuration** page.

ArubaOS 5.0 | User Guide 802.1x Authentication | 287

7. In the AP Group list, click **Edit** for the second-floor.

8. In the Profiles list, select Wireless LAN, then select Virtual AP.

9. Select **guest** from the Add a profile drop-down menu. Click **Add**.

10. Click **Apply**.

Using the CLI to configure the guest WLAN

wlan ssid-profile guest

essid guest

wepkey1 aaaaaaaaaa

opmode static-wep

wlan virtual-ap guest

vlan 63

ssid-profile guest

ap-group first-floor

virtual-ap guest

ap-group second-floor

virtual-ap guest

Non-Guest WLANs

You create and configure the SSID profile “WLAN-01” with the ESSID “WLAN-01” and WPA TKIP

encryption. You need to create and configure two virtual AP profiles: one with VLAN 60 for the first-floor

AP group and the other with VLAN 61 for the second-floor AP group. Each virtual AP profile references the

SSID profile “WLAN-01” and the previously-configured AAA profile “aaa\_dot1x”.

Using the WebUI to configure the non-guest WLANs

1. Navigate to the **Configuration > Wireless > AP Configuration** page.

2. In the AP Group list, click **Edit** for the first-floor.

3. In the Profiles list, select Wireless LAN, then select Virtual AP.

4. To configure the WLAN-01\_first-floor virtual AP:

a. Select NEW from the Add a profile drop-down menu. Enter **WLAN-01\_first-floor**, and click **Add**.

b. In the Profile Details entry for the WLAN-01\_first-floor virtual AP profile, select the **aaa\_dot1x** AAA

profile you previously configured. A pop-up window displays the configured AAA profile parameters.

Click **Apply** in the pop-up window.

c. From the SSID profile drop-down menu, select NEW. A pop-up window allows you to configure the

SSID profile.

d. Enter **WLAN-01** for the name of the SSID profile.

e. For Network Name, enter **WLAN-01**.

f. For Network Authentication, select **WPA**.

g. Click **Apply** in the pop-up window.

h. At the bottom of the Profile Details page, click **Apply**.

5. Click on the WLAN-01\_first-floor virtual AP name in the Profiles list or in Profile Details to display

configuration parameters.

a. Make sure Virtual AP enable is selected.

b. For VLAN, select 60.

c. Click **Apply**.

6. Navigate to the **Configuration > Wireless > AP Configuration** page.

288 | 802.1x Authentication ArubaOS 5.0 | User Guide

7. In the AP Group list, click **Edit** for the second-floor.

8. In the Profiles list, select Wireless LAN, then select Virtual AP.

9. To configure the WLAN-01\_second-floor virtual AP:

a. Select NEW from the Add a profile drop-down menu. Enter **WLAN-second-floor**, and click **Add**.

b. In the Profile Details entry for the virtual AP profile, select **aaa\_dot1x** from the AAA profile dropdown

menu. A pop-up window displays the configured AAA profile parameters. Click **Apply** in the

pop-up window.

c. From the SSID profile drop-down menu, select **WLAN-01**. A pop-up window displays the configured

SSID profile parameters. Click **Apply** in the pop-up window.

d. At the bottom of the Profile Details page, click **Apply**.

10. Click on the new virtual AP name in the Profiles list or in Profile Details to display configuration

parameters.

a. Make sure Virtual AP enable is selected.

b. For VLAN, select 61.

c. Click **Apply**.

Using the CLI to configure the non-guest WLANs

wlan ssid-profile WLAN-01

essid WLAN-01

opmode wpa-tkip

wlan virtual-ap WLAN-01\_first-floor

vlan 60

aaa-profile aaa\_dot1x

ssid-profile WLAN-01

wlan virtual-ap WLAN-01\_second-floor

vlan 61

aaa-profile aaa\_dot1x

ssid-profile WLAN-01

ap-group first-floor

virtual-ap WLAN-01\_first-floor

ap-group second-floor

virtual-ap WLAN-01\_second-floor

Authentication with the Controller’s Internal Database

In the following example:

􀁺 The controller’s internal database provides user authentication.

􀁺 The authentication type is WPA. From the 802.1x authentication exchange, the client and the controller

derive dynamic keys to encrypt data transmitted on the wireless network.

Configuring Policies and Roles

Create the following policies and user roles:

􀁺 The **student** policy prevents students from using telnet, POP3, FTP, SMTP, SNMP, or SSH to the wired

portion of the network. The **student** policy is mapped to the **student** user role.

􀁺 The **faculty** policy is similar to the **student** policy, however faculty members are allowed to use POP3

and SMTP for VPN remote access from home. (Students are not permitted to use VPN remote access.)

The **faculty** policy is mapped to the **faculty** user role.

ArubaOS 5.0 | User Guide 802.1x Authentication | 289

􀁺 The **guest** policy permits only access to the Internet (via HTTP or HTTPS) and only during daytime

working hours. The **guest** policy is mapped to the **guest** user role.

􀁺 The **allowall** policy, a predefined policy, allows unrestricted access to the network. The **allowall** policy

is mapped to both the **sysadmin** user role and the **computer** user role.

Using the Web to create the student policy and role

1. Navigate to the **Configuration > Security > Access Control > Policies** page. Select **Add** to add the

student policy.

2. For Policy Name, enter **student**.

3. For Policy Type, select **IPv4 Session**.

4. Under Rules, select **Add** to add rules for the policy.

a. Under Source, select **user**.

b. Under Destination, select **alias**.

c. Under the alias selection, click **New**. For Destination Name, enter “Internal Network”. Click **Add** to

add a rule. For Rule Type, select **network**. For IP Address, enter 10.0.0.0. For Network Mask/Range,

enter 255.0.0.0. Click **Add** to add the network range. Repeat these steps to add the network range

172.16.0.0 255.255.0.0. Click **Done**. The alias “Internal Network” appears in the Destination menu.

d. Under Destination, select Internal Network.

e. Under Service, select **service**. In the Service scrolling list, select **svc-telnet**.

f. Under Action, select **drop**.

g. Click **Add**.

5. Under Rules, click **Add**.

a. Under Source, select **user**.

b. Under Destination, select **alias**. Then select Internal Network.

c. Under Service, select **service**. In the Service scrolling list, select **svc-pop3**.

d. Under Action, select **drop**.

e. Click **Add**.

6. Repeat steps 4A-E to create rules for the following services: svc-ftp, svc-smtp, svc-snmp, and svc-ssh.

7. Click **Apply**.

8. Click the **User Roles** tab. Click **Add** to create the student role.

9. For Role Name, enter **student**.

10. Under Firewall Policies, click **Add**. In Choose from Configured Policies, select the student policy you

previously created. Click **Done**.

11. Click **Apply**.

Using the WebUI to create the faculty policy and role

1. Navigate to the **Configuration > Security > Access Control > Policies** page. Click **Add** to add the

faculty policy.

2. For Policy Name, enter **faculty**.

3. For Policy Type, select **IPv4 Session**.

4. Under Rules, click **Add** to add rules for the policy.

N O T E

The following step defines an alias representing all internal network addresses. Once defined, you can use the alias

for other rules and policies.

290 | 802.1x Authentication ArubaOS 5.0 | User Guide

a. Under Source, select **user**.

b. Under Destination, select alias, then select **Internal Network**.

c. Under Service, select **service**. In the Service scrolling list, select **svc-telnet**.

d. Under Action, select **drop**.

e. Click **Add**.

f. Repeat steps A-E to create rules for the following services: svc-ftp, svc-snmp, and svc-ssh.

5. Click **Apply**.

6. Select the **User Roles** tab. Click **Add** to create the faculty role.

7. For Role Name, enter **faculty**.

8. Under **Firewall Policies**, click **Add**. In Choose from Configured Policies, select the faculty policy you

previously created. Click **Done**.

Using the WebUI to create the guest policy and role

1. Navigate to the **Configuration > Security > Access Control > Time Ranges** page to define the time

range “working-hours”. Click **Add**.

a. For Name, enter **working-hours**.

b. For Type, select **Periodic**.

c. Click **Add**.

d. For Start Day, click **Weekday**.

e. For Start Time, enter **07:30**.

f. For End Time, enter **17:00**.

g. Click **Done**.

h. Click **Apply**.

2. Click the **Policies** tab. Click **Add** to add the guest policy.

3. For Policy Name, enter **guest**.

4. For Policy Type, select **IPv4 Session**.

5. Under Rules, click **Add** to add rules for the policy.

To create rules to permit access to DHCP and DNS servers during working hours:

a. Under Source, select **user**.

b. Under Destination, select **host**. In Host IP, enter **10.1.1.25**.

c. Under Service, select **service**. In the Service scrolling list, select **svc-dhcp**.

d. Under Action, select **permit**.

e. Under Time Range, select **working-hours**.

f. Click **Add**.

g. Repeat steps A-F to create a rule for svc-dns.

To create a rule to deny access to the internal network:

a. Under Source, select **user**.

b. Under Destination, select **alias**. Select **Internal Network**.

c. Under Service, select **any**.

d. Under Action, select **drop**.

e. Click **Add**.

To create rules to permit HTTP and HTTPS access during working hours:

ArubaOS 5.0 | User Guide 802.1x Authentication | 291

a. Under Source, select **user**.

b. Under Destination, select **any**.

c. Under Service, select service. In the Services scrolling list, select **svc-http**.

d. Under Action, select **permit**.

e. Under Time Range, select **working-hours**.

f. Click **Add**.

g. Repeat steps A-F for the svc-https service.

To create a rule that denies the user access to all destinations and all services:

a. Under Source, select **user**.

b. Under Destination, select **any**.

c. Under Service, select **any**.

d. Under Action, select **drop**.

e. Click **Add**.

6. Click **Apply**.

7. Click the **User Roles** tab. Click **Add** to create the guest role.

8. For Role Name, enter **guest**.

9. Under **Firewall Policies**, click **Add**. In Choose from Configured Policies, select the guest policy you

previously created. Click **Done**.

Using the WebUI to create the sysadmin role

1. Navigate to **Configuration > Security > Access Control > User Roles** page. Click **Add** to create the

sysadmin role.

2. For Role Name, enter **sysadmin**.

3. Under Firewall Policies, click **Add**. In Choose from Configured Policies, select the predefined **allowall**

policy. Click **Done**.

4. Click **Apply**.

Using the WebUI to create the computer role

1. Navigate to **Configuration > Security > Access Control > User Roles** page. Click **Add** to create the

computer role.

2. For Role Name, enter **computer**.

3. Under Firewall Policies, click **Add**. In Choose from Configured Policies, select the predefined **allowall**

policy. Click **Done**.

4. Click **Apply**.

Using the CLI to create an alias for the internal network

netdestination “Internal Network”

network 10.0.0.0 255.0.0.0

network 172.16.0.0 255.255.0.0

Using the CLI to create the student role

ip access-list session student

user alias “Internal Network” svc-telnet deny

user alias “Internal Network” svc-pop3 deny

user alias “Internal Network” svc-ftp deny

user alias “Internal Network” svc-smtp deny

user alias “Internal Network” svc-snmp deny

292 | 802.1x Authentication ArubaOS 5.0 | User Guide

user alias “Internal Network” svc-ssh deny

user-role student

session-acl student

session-acl allowall

Using the CLI to create the faculty role

ip access-list session faculty

user alias “Internal Network” svc-telnet deny

user alias “Internal Network” svc-ftp deny

user alias “Internal Network” svc-snmp deny

user alias “Internal Network” svc-ssh deny

user-role faculty

session-acl faculty

session-acl allowall

Using the CLI to create the guest role

time-range working-hours periodic

weekday 07:30 to 17:00

ip access-list session guest

user host 10.1.1.25 svc-dhcp permit time-range working-hours

user host 10.1.1.25 svc-dns permit time-range working-hours

user alias “Internal Network” any deny

user any svc-http permit time-range working-hours

user any svc-https permit time-range working-hours

user any any deny

user-role guest

session-acl guest

Using the CLI to create the sysadmin role

user-role sysadmin

session-acl allowall

Using the CLI to create the computer role

user-role computer

session-acl allowall

Configuring the Internal Database

Configure the internal database with the username, password, and role (student, faculty, or sysadmin) for

each user. There is a default **internal** server group that includes the internal database. For the internal

server group, configure a server derivation rule that assigns the role to the authenticated client.

Using the WebUI to configure the internal database

1. Navigate to the **Configuration > Security > Authentication > Servers** page.

2. In the Servers list, select Internal DB.

3. Under Users, click **Add User** to add users.

4. For each user, enter a username and password.

5. Select the Role for each user (if a role is not specified, the default role is guest).

6. Select the expiration time for the user account in the internal database.

7. Click **Apply**.

ArubaOS 5.0 | User Guide 802.1x Authentication | 293

Using the WebUI to configure a server rule for the internal database

1. Navigate to the **Configuration > Security > Authentication > Servers** page.

2. Select **Server Group** to display the Server Group list.

3. Select the **internal** server group.

4. Under Server Rules, click **New** to add a server derivation rule.

a. For Condition, enter Role.

b. Select value-of from the drop-down menu.

c. Select Set Role from the drop-down menu.

d. Click **Add**.

5. Click **Apply**.

Using the CLI to configure the internal database

local-userdb add username <*user>* password <*password>*

Using the CLI to configure a server rule for the internal database

aaa server-group internal

set role condition Role value-of

Configure 802.1x Authentication

An AAA profile specifies the 802.1x authentication profile and 802.1x server group to be used for

authenticating clients for a WLAN. The AAA profile also specifies the default user role for 802.1x

authentication.

For this example, you enable both 802.1x authentication and termination on the controller.

Using the WebUI to configure 802.1x authentication

1. Navigate to the **Configuration > Security > Authentication > L2 Authentication** page. In the

profiles list, select 802.1x Authentication Profile.

a. In the Instance list, enter **dot1x**, then click **Add**.

b. Select the dot1x profile you just created.

c. Select **Termination**.

d. Click **Apply**.

2. Select the **AAA Profiles** tab.

a. In the AAA Profiles Summary, click **Add** to add a new profile.

b. Enter **aaa\_dot1x**, then click **Add**.

c. Select the aaa\_dot1x profile you just created.

d. For 802.1x Authentication Default Role, select **faculty**.

e. Click **Apply**.

3. In the Profiles list (under the aaa\_dot1x profile you just created), select 802.1x Authentication Profile.

N O T E

Use the privileged mode in the CLI to configure users in the controller’s internal database.

N O T E

The defaults for EAP Method and Inner EAP Method are EAP-PEAP and EAP-MSCHAPv2, respectively.

294 | 802.1x Authentication ArubaOS 5.0 | User Guide

a. Select the dot1x profile from the 802.1x Authentication Profile drop-down menu.

b. Click **Apply**.

4. In the Profiles list (under the aaa\_dot1x profile you just created), select 802.1x Authentication Server

Group.

a. Select the **internal** server group.

b. Click **Apply**.

Using the CLI to configure 802.1x authentication

aaa authentication dot1x dot1x

termination enable

aaa profile aaa\_dot1x

dot1x-default-role student

authentication-dot1x dot1x

dot1x-server-group internal

Configure VLANs

In this example, wireless clients are assigned to either VLAN 60 or 61 while guest users are assigned to

VLAN 63. VLANs 60 and 61 split users into smaller IP subnetworks, improving performance by decreasing

broadcast traffic. The VLANs are internal to the Aruba controller only and do not extend into other parts of

the wired network. The clients’ default gateway is the Aruba controller, which routes traffic out to the

10.1.1.0 subnetwork.

You configure the VLANs, assign IP addresses to each VLAN, and establish the “helper address” to which

client DHCP requests are forwarded.

Using the WebUI to configure VLAN

1. Navigate to the **Configuration > Network > VLAN** page. Click **Add** to add VLAN 60.

a. For VLAN ID, enter **60**.

b. Click **Apply**.

c. Repeat steps A and B to add VLANs 61 and 63.

2. To configure IP parameters for the VLANs, navigate to the **Configuration > Network > IP > IP**

**Interfaces** page.

a. Click **Edit** for VLAN 60.

b. For IP Address, enter **10.1.60.1**.

c. For Net Mask, enter **255.255.255.0**.

d. Under DHCP Helper Address, click **Add**. Enter **10.1.1.25** and click **Add**.

e. Click **Apply**.

3. In the IP Interfaces page, click **Edit** for VLAN 61.

a. For IP Address, enter **10.1.61.1**.

b. For Net Mask, enter **255.255.255.0**.

c. Under DHCP Helper Address, click **Add**. Enter **10.1.1.25** and click **Add**.

d. Click **Apply**.

4. In the IP Interfaces page, click **Edit** for VLAN 63.

a. For IP Address, enter **10.1.63.1**.

b. For Net Mask, enter **255.255.255.0**.

c. Under DHCP Helper Address, click **Add**. Enter **10.1.1.25** and click **Add**.

ArubaOS 5.0 | User Guide 802.1x Authentication | 295

d. Click **Apply**.

5. Select the **IP Routes** tab.

a. For Default Gateway, enter **10.1.1.254**.

b. Click **Apply**.

Using the CLI to configure VLANs

vlan 60

interface vlan 60

ip address 10.1.60.1 255.255.255.0

ip helper-address 10.1.1.25

vlan 61

interface vlan 61

ip address 10.1.61.1 255.255.255.0

ip helper-address 10.1.1.25

vlan 63

interface vlan 63

ip address 10.1.63.1 255.255.255.0

ip helper-address 10.1.1.25

ip default-gateway 10.1.1.254

Configure the WLANs

In this example, default AP parameters for the entire network are as follows: the default ESSID is WLAN-01

and the encryption mode is TKIP. A second ESSID called “guest” has the encryption mode set to static WEP

with a configured WEP key.

In this example, the non-guest clients that associate to an AP are mapped into one of two different user

VLANs. The initial AP to which the client associates determines the VLAN: clients that associate to APs in

the first floor of the building are mapped to VLAN 60 and clients that associate to APs in the second floor of

the building are mapped to VLAN 61. Therefore, the APs in the network are segregated into two AP groups,

named “first-floor” and “second-floor”. (See “AP Groups” on page 109 for information about creating AP

groups.) The guest clients are mapped into VLAN 63.

Guest WLAN

You create and configure the virtual AP profile “guest” and apply the profile to each AP group. The “guest”

virtual AP profile contains the SSID profile “guest” which configures static WEP with a WEP key.

Using the WebUI to configure the WLAN

1. Navigate to the **Configuration > Wireless > AP Configuration** page.

2. In the AP Group list, select first-floor.

3. In the Profiles list, select Wireless LAN then select Virtual AP.

4. To configure the guest virtual AP:

a. Select NEW from the Add a profile drop-down menu. Enter **guest** for the name of the virtual AP

profile, and click **Add**.

b. In the Profile Details entry for the guest virtual AP profile, select NEW from the SSID profile dropdown

menu. A pop-up window allows you to configure the SSID profile.

c. Enter **guest** for the name of the SSID profile.

d. Enter **guest** for the Network Name.

296 | 802.1x Authentication ArubaOS 5.0 | User Guide

e. For Network Authentication, select **None**.

f. For Encryption, select **WEP**.

g. Enter the WEP key.

h. Click **Apply**.

i. Under Profile Details, click **Apply**.

5. Click on the guest virtual AP name in the Profiles list or in Profile Details to display configuration

parameters.

a. Make sure Virtual AP enable is selected.

b. For VLAN, select **63**.

c. Click **Apply**.

6. Navigate to the **Configuration > Wireless > AP Configuration** page.

7. In the AP Group list, select second-floor.

8. In the Profiles list, select Wireless LAN, then select Virtual AP.

9. Select **guest** from the Add a profile drop-down menu. Click **Add**.

10. Click **Apply**.

Using the CLI to configure the guest WLAN

wlan ssid-profile guest

essid guest

wepkey1 aaaaaaaaaa

opmode static-wep

wlan virtual-ap guest

vlan 63

ssid-profile guest

ap-group first-floor

virtual-ap guest

ap-group second-floor

virtual-ap guest

Non-Guest WLANs

You create and configure the SSID profile “WLAN-01” with the ESSID “WLAN-01” and WPA TKIP

encryption. You need to create and configure two virtual AP profiles: one with VLAN 60 for the first-floor

AP group and the other with VLAN 61 for the second-floor AP group. Each virtual AP profile references the

SSID profile “WLAN-01” and the previously-configured AAA profile “aaa\_dot1x”.

Using the WebUI to configure the non-guest WLANs

1. Navigate to the **Configuration > Wireless > AP Configuration** page.

2. In the AP Group list, select first-floor.

3. In the Profiles list, select Wireless LAN, then select Virtual AP.

4. To configure the WLAN-01\_first-floor virtual AP:

a. Select NEW from the Add a profile drop-down menu. Enter **WLAN-01\_first-floor**, and click **Add**.

b. In the Profile Details entry for the WLAN-01\_first-floor virtual AP profile, select **aaa\_dot1x** from the

AAA Profile drop-down menu. A pop-up window displays the configured AAA parameters. Click

**Apply** in the pop-up window.

c. From the SSID profile drop-down menu, select NEW. A pop-up window allows you to configure the

SSID profile.

ArubaOS 5.0 | User Guide 802.1x Authentication | 297

d. Enter **WLAN-01** for the name of the SSID profile.

e. Enter **WLAN-01** for the Network Name.

f. Select **WPA** for Network Authentication.

g. Click **Apply** in the pop-up window.

h. At the bottom of the Profile Details page, click **Apply**.

5. Click on the WLAN-01\_first-floor virtual AP profile name in the Profiles list or in Profile Details to

display configuration parameters.

a. Make sure Virtual AP enable is selected.

b. For VLAN, select 60.

c. Click **Apply**.

6. Navigate to the **Configuration > Wireless > AP Configuration** page.

7. In the AP Group list, select second-floor.

8. In the Profiles list, select Wireless LAN then select Virtual AP.

9. To create the WLAN-01\_second-floor virtual AP:

a. Select NEW from the Add a profile drop-down menu. Enter **WLAN-01\_second-floor**, and click **Add**.

b. In the Profile Details entry for the virtual AP profile, select **aaa\_dot1x** from the AAA Profile dropdown

menu. A pop-up window displays the configured AAA profile parameters. Click **Apply** in the

pop-up window.

c. From the SSID profile drop-down menu, select **WLAN-01**. a pop-up window displays the configured

SSID profile parameters. Click **Apply** in the pop-up window.

d. At the bottom of the Profile Details page, click **Apply**.

10. Click on the WLAN-01\_second-floor virtual AP profile name in the Profiles list or in Profile Details to

display configuration parameters.

a. Make sure Virtual AP enable is selected.

b. For VLAN, select 61.

c. Click **Apply**.

Using the CLI to configure the non-guest WLANs

wlan ssid-profile WLAN-01

essid WLAN-01

opmode wpa-tkip

wlan virtual-ap WLAN-01\_first-floor

vlan 60

aaa-profile aaa\_dot1x

ssid-profile WLAN-01

wlan virtual-ap WLAN-01\_second-floor

vlan 61

aaa-profile aaa\_dot1x

ssid-profile WLAN-01

ap-group first-floor

virtual-ap WLAN-01\_first-floor

ap-group second-floor

virtual-ap WLAN-01\_second-floor

298 | 802.1x Authentication ArubaOS 5.0 | User Guide

Advanced Configuration Options for 802.1x

This section describes advanced configuration options for 802.1x authentication.

Reauthentication with Unicast Key Rotation

When enabled, unicast and multicast keys are updated after each reauthorization. It is a best practice to

configure the time intervals for reauthentication, multicast key rotation, and unicast key rotation to be at

least 15 minutes. Make sure these intervals are mutually prime, and the factor of the unicast key rotation

interval and the multicast key rotation interval is less than the reauthentication interval.

The following is an example of the parameters you can configure for reauthentication with unicast and

multicast key rotation:

􀁺 Reauthentication: Enabled

􀁺 Reauthentication Time Interval: 6011 Seconds

􀁺 Multicast Key Rotation: Enabled

􀁺 Multicast Key Rotation Time Interval:1867 Seconds

􀁺 Unicast Key Rotation: Enabled

􀁺 Unicast Key Rotation Time Interval: 1021 Seconds

Using the WebUI to configure reauthentication with unicast key rotation

1. Navigate to the **Configuration > Security > Authentication > L2 Authentication** page.

2. Select 802.1x Authentication Profile, then select the name of the profile you want to configure.

3. Select the **Advanced** tab. Enter the following values:

􀂄 Reauthentication Interval: 6011

􀂄 Multicast Key Rotation Time Interval: 1867

􀂄 Unicast Key Rotation Time Interval: 1021

􀂄 Multicast Key Rotation: (select)

􀂄 Unicast Key Rotation: (select)

􀂄 Reauthentication: (select)

4. Click **Apply**.

Using the CLI to configure reauthentication with unicast key rotation

aaa authentication dot1x *profile*

reauthentication

timer reauth-period 6011

unicast-keyrotation

timer ukey-rotation-period 1021

multicast-keyrotation

timer mkey-rotation-period 1867