AP6522 Client Bridge
HOW TO GUIDE

Part No. TME-08-2014-11

Table of Contents

1. Introduction

Client bridges provide simple, cost-effective Wireless LAN connectivity for printers, point-of-sale and Ethernet enabled devices. Client bridges have been traditionally deployed in enterprises to provide Ethernet connectivity where it is cost prohibitive or infeasible to deploy physical cabling or where mobility is required.

In WiNG **[NEED RELEASE]** and above Zebra Solutions introduces client bridge radio support for the AP6522 Independent Access Points as a next generation replacement for the CB3000 Wireless Bridge Adaptor. The AP6522 includes a dual band-unlocked 2x2:2 MIMO radio for Wireless LAN connectivity providing increased speed, performance and signal integrity. The AP6522 also includes an integrated Gigabit Ethernet port providing faster throughput and higher transfer rates to support the current generation of wired hosts and bandwidth hungry applications. Finally the AP6522 client bridge is based on WiNG software allowing enterprises to leverage advances features such as centralized management and port-based security.

The following table provides a summary of the AP6522 client bridge features and benefits:

Features	Benefit
Form Factor	Integrated and External Antenna options.
IEEE 802.11a/b/g/n	Dual band support providing seamless connectivity to Zebra and third-party 802.11n and legacy 802.11a/b/g Wireless LAN Networks.
Flexible Power Options	AC power or IEEE 802.3af Power over Ethernet (PoE).
Protocol Agnostic	Support for current IPv4 and IPv6 protocols in addition to legacy protocols such as LLC required by older point-of-sale systems.
Secure	Support for current 802.11i standards for Wireless LAN encryption and authentication, integrated firewall for policy enforcement and IEEE 802.1X port-based security for authenticating wired hosts.
Management	Full centralized management, configuration and monitoring with scaling up to 10,240 client bridges.

Table 1 – AP6522 Client Bridge Benefits

The specifications for the AP6522 Independent Access Point can be viewed from the Zebra Solutions website here.

1.1 Radio Configuration Parameters

The following table provides an overview of the new client bridge radio configuration parameters which can be defined in the Profile or Device context for radio 2 on the AP6522 Access Point. Client bridge functionality is not supported for radio 1:

Parameter	Description
rf-mode bridge	Configures Radio 2 to operate as a Bridge Radio. Default is 5GHz
bridge ssid <ssid-name>	The SSID of the Wireless LAN Radio 2 will connect to.
bridge authentication-type [eap | none]	Authentication of the Wireless LAN Radio 2 will connect to: ▪ ***eap*** – Use EAP Authentication (PEAP MSCHAPv2) ▪ ***none*** – Use no Authentication
bridge encryption-type [none | tkip | ccmp]	Encryption of the Wireless LAN Radio 2 will connect to: ▪ ***none*** – Use an Open Network (No Encryption) ▪ ***tkip*** – Use WPA / WPA2 TKIP Encryption ▪ ***ccmp*** – Use WPA / WPA2 CCMP Encryption
bridge wpa-wpa2 psk <pre-shared-key>	The WPA / WPA2 pre-shared key.
bridge eap type [peap-mschapv2 | tls]	Determines the EAP method used for EAP authentication: ▪ ***peap-mschapv2*** – Selects PEAP-MSCHAPv2 ▪ ***tls*** – Selects EAP-TLS Default is peap-mschapv2.
bridge eap username <username>	EAP Username: ▪ ***PEAP-MSCHAPv2*** – PEAP username (example *client-bridge*) ▪ ***EAP-TLS*** – Username in the CN field of the installed PKCS #12 client certificate (example *client-bridge@example.com*)
bridge eap password <password>	EAP Password: ▪ ***PEAP-MSCHAPv2*** – PEAP password (example *hellomoto*) ▪ ***EAP-TLS*** – PKCS #12 certificate secret
bridge channel-list [2.4GHz | 5GHz] <list>	Channel list Radio 2 uses to scan for Portal Access Points: ▪ ***2.4GHz*** – 2.4GHz comma separated channel list ▪ ***5GHz*** – 5GHz comma separated channel list

Parameter	Description
bridge roam-criteria missed-beacons <1-60>	Number consecutive missed beacons before roaming. Default 20 beacons.
bridge roam-criteria rssi-threshold <-128 to -40>	Minimum signal strength for the bridge connection to be maintained before roaming. Default -75 RSSI.
bridge inactivity-timeout <10-655350>	Inactivity timeout in seconds. If a frame is not received from a wired client for this amount of time, the client is deleted. Default 600 seconds.
bridge keepalive interval <0-36000>	Interval in seconds at which keepalive frames are to be sent on behalf of bridged clients. Default is 300 seconds.
bridge keepalive frame-type [null-data | wmnp]	Defines the frame type transmitted as a keepalive: ▪ ***null-data*** **–**Transmits 802.11 NULL Data Frame ▪ ***wnmp*** **–** Transmits WMNP multicast packet Default is null-data.
bridge max-clients <1-14>	Maximum number of wired clients bridged by Radio 2. Default 14.

Table 1.1 – Client Bridge Radio 2 Parameters

By default all permitted 2.4GHz and 5GHz channels are scanned by default. You can prevent the scanning of a particular band by issuing the ***bridge channel-list 2.4GHz*** *or* ***bridge channel-list 5GHz*** *command with no channel list assigned.*

Pre-shared keys are only valid when the ***authentication-type*** *is set to* ***none*** *and the* ***encryption-type*** *is set to* ***tkip*** *or* ***ccmp****.*

Per the 802.11n amendment, 802.11n data-rates can only be achieved when the ***encryption-type*** *is set to* ***ccmp*** *or* ***none****.*

For simplify and ease of deployment the client bridge radio will automatically map the bridged traffic to ***VLAN 1****.*

1.2 Bridging Architecture

When an AP6522 Access Point is configured as a client bridge, radio 2 will associate and authenticate to the defined infrastructure Wireless LAN in the same manner as a Wireless Client. The Wireless LAN infrastructure will map the client bridge session to a VLAN allowing the AP6522 client bridge to communicate with permitted hosts over the infrastructure Wireless LAN. An AP6522 client bridge can connect to WPA/WPA2 infrastructure Wireless LANs using pre-shared keys (PSK) or EAP. An AP6522 client bridge can also connect Open Wireless LAN if desired.

Once an AP6522 client bridge is connected to an infrastructure Wireless LAN, the AP6522 client bridge can switch frames between radio 2 and wired host(s) connected to its Ge1 port. For simplicity and ease of deployment all traffic by default is bridged to VLAN 1 which is assigned to both the client bridge radio 2 and Ge1 port by default. Each AP6522 client bridge will also have a dynamic IPv4 address assigned to VLAN 1 which is used for remote management and adoption.

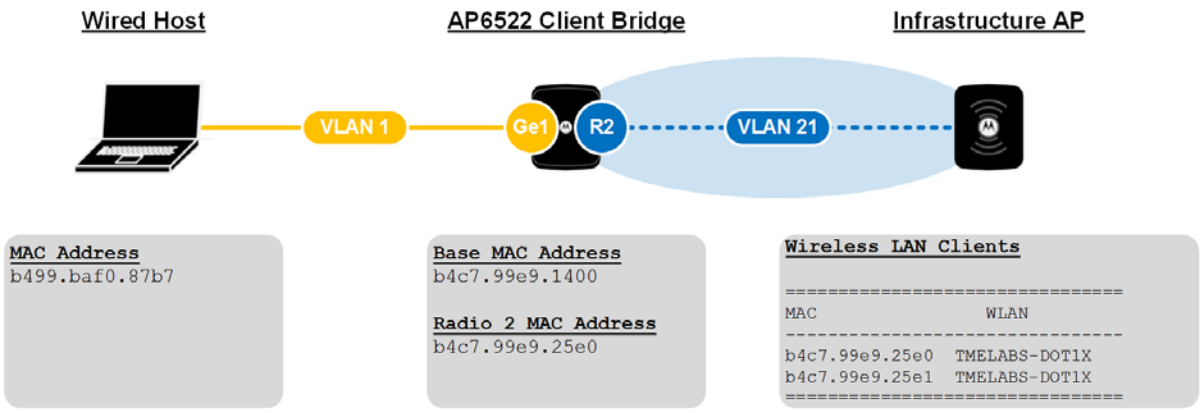


Figure 1.2 – Bridging Architecture

One unique characteristic of a client bridge is how the wired host(s) MAC addresses are presented to the infrastructure Wireless LAN. Each AP6522 client bridge connected to the infrastructure Wireless LAN will display as an active session in the same way as a Wireless Client. The MAC address used for the session being the base MAC address assigned to radio 2 on each AP6522 client bridge.

Each wired host connected to the Ge1 port on the AP6522 client bridge will also be displayed as an active session on the infrastructure Wireless LAN. However the wired host(s) real MAC address will not be displayed. The MAC addresses of each wired host is masqueraded and translated to a unique MAC address that is based on the AP6522 client bridges radio 2 base MAC. Each wired host connected to the AP6522 client bridges Ge1 port is assigned a unique MAC address based on the radio 2 MAC address + 1. Each AP6522 client bridge can support a maximum of 16 wired hosts which corresponds to the pool of MAC addresses allocated to radio 2.

2. Configuration Examples

2.1 Standalone Deployments

The following section provides an overview of the configuration steps required to deploy AP6522 Access Points as standalone client bridges. All configuration will be performed directly on the AP 6522 Access Points using the command line interface (CLI) using the following Policies, RF Domain and Profile:

1. Default Firewall Policy
2. Default Management Policy
3. Default RF Domain
4. Default Profile

2.1.1 Default Firewall Policy

Firewall Policies determine which firewall services are enabled on each WiNG 5 device. By default each WiNG 5 device is assigned the default Firewall Policy which is automatically mapped to default and user defined Profiles.

As the AP6522 will be deployed as a client bridge and not an infrastructure Access Point, all firewall services will be disabled. Stateful packet inspection and policy enforcement will be primarily provided by the Wireless LAN infrastructure and not the AP6522 client bridges. It is possible however to define and assign IP or MAC Access Control Lists (ACLs) to the Ge1 port if required.

The following demonstrates how to modify the default Firewall Policy following recommended best practices for standalone client bridge deployments:

1	**Access the default Firewall Policy configuration context:**

```
AP6522-CB1(config)# firewall-policy default
```

2	**Disable DoS detection:**

```
AP6522-CB1(config-fw-policy-default)# no ip dos
```

3	**Disable IP MAC Conflict detection:**

```
AP6522-CB1(config-fw-policy-default)# no ip-mac conflict
AP6522-CB1(config-fw-policy-default)# no ip-mac routing conflict
```

4	**Disable Layer 2 Stateful Packet Inspection:**

```
AP6522-CB1(config-fw-policy-default)# no stateful-packet-inspection-l2
```

5	**Disable the Layer 3 Firewall:**

```
AP6522-CB1(config-fw-policy-default)# no firewall enable
```

6	**Exit then Commit and Write the Changes:**

```
AP6522-CB1(config-fw-policy-default)# exit
AP6522-CB1(config)# commit write
```

Resulting Configuration Changes:

```
!
firewall-policy default
 no ip dos smurf
 no ip dos twinge
 no ip dos invalid-protocol
 no ip dos router-advt
 no ip dos router-solicit
 no ip dos option-route
 no ip dos ascend
 no ip dos chargen
 no ip dos fraggle
 no ip dos snork
 no ip dos ftp-bounce
 no ip dos tcp-intercept
 no ip dos broadcast-multicast-icmp
 no ip dos land
 no ip dos tcp-xmas-scan
 no ip dos tcp-null-scan
 no ip dos winnuke
 no ip dos tcp-fin-scan
 no ip dos udp-short-hdr
 no ip dos tcp-post-syn
 no ip dos tcphdrfrag
 no ip dos ip-ttl-zero
 no ip dos ipspoof
 no ip dos tcp-bad-sequence
 no ip dos tcp-sequence-past-window
 no ip-mac conflict
 no ip-mac routing conflict
 no firewall enable
 no stateful-packet-inspection-l2
!
```

2.1.2 Default Management Policy

Management Policies determine which management services and administrative user accounts are enabled on each WiNG 5 device. Management Policies also determine how the administrative user accounts are authenticated and authorized. By default each WiNG 5 device is assigned the default Management Policy which is automatically mapped to default and user defined Profiles.

As a general best practice it is recommended that you disable all un-necessary management services on the AP6522 client bridges and if enabled the management service should be secured. For standalone client bridge deployments it may be desirable to enable SSHv2, HTTPS and SNMPv3 management services as no Centralized Controller is deployed. At a minimum SSHv2 should be enabled to provide remote management Access over the intermediate network and remote management access should be restricted to specific hosts or subnets.

The following demonstrates how to modify the default Management Policy following recommended best practices for standalone client bridge deployments:

1 Access the default Management Policy configuration context:

```
AP6522-CB1(config)# management-policy default
```

2 Disable un-necessary management services:

```
AP6522-CB1(config-management-policy-default)# no https server
AP6522-CB1(config-management-policy-default)# no snmp-server manager all
```

3 Restrict remote management access to a specific Host(s) or Subnet(s):

```
AP6522-CB1(config-management-policy-default)# restrict-access subnet 192.168.10.0/24
```

4 Exit then Commit and Write the Changes:

```
AP6522-CB1(config-management-policy-default)# exit
AP6522-CB1(config)# commit write
```

Resulting Configuration Changes:

```
!
management-policy default
 no http server
 ssh
 user admin password 1 <obfuscated-password> role superuser access all
 no snmp-server manager v3
 snmp-server community 0 public ro
 snmp-server user snmptrap v3 encrypted des auth md5 0 Zebra
 snmp-server user snmpmanager v3 encrypted des auth md5 0 Zebra
 restrict-access subnet 192.168.10.0/24
!
```

2.1.3 Default RF Domain

RF Domains determine regulatory a regional operating parameters for the AP6522 client bridges and in managed deployments are also used for organization and visualization. By default each WiNG 5 device is assigned to a default RF Domain which is automatically mapped to the devices configuration.

For each AP6522 client bridge the default RF Domain must be modified and a ISO 3166 Country Code assigned. The Country Code must be compatible with the AP6522 hardware SQU you purchased as each AP6522 SQU is locked to a specific region. As a general best practice it is also recommended that you define a Location, Contact and TimeZone to simplify the identification and management of the client bridges.

The following demonstrates how to modify the default RF Domain to following recommended best practices for standalone client bridge deployments:

1 Access the default RF Domain configuration context:

```
AP6522-CB1(config)# rf-domain default
```

2 Define a Location, Contact, Country Code and TimeZone:

```
AP6522-CB1(config-rf-domain-default)# location “Johnson City TN”
AP6522-CB1(config-rf-domain-default)# contact “kmarshall@Zebrasolutions.com”
AP6522-CB1(config-rf-domain-default)# country-code us
AP6522-CB1(config-rf-domain-default)# timezone EST5EDT
```

3 Exit then Commit and Write the Changes:

```
AP6522-CB1(config-rf-domain-default)# exit
AP6522-CB1(config)# commit write
```

Resulting Configuration Changes:

```
!
rf-domain default
 location “Johnson City TN”
 contact kmarshall@Zebrasolutions.com
 timezone EST5EDT
 country-code us
!
```

2.1.4 Default AP6522 Profile

Profiles assign common configuration parameters to groups of managed WiNG5 devices and are model specific. By default each WiNG 5 device is assigned to a default Profile which is automatically mapped to the devices configuration.

For each AP6522 client bridge the default AP6522 Profile will be modified to assign DNS and NTP parameters in addition to configuring radio 2 as a client bridge radio to connect to a Wireless LAN serviced by the infrastructure Access Points. Examples will be provided to configure the client bridge radio to connect to pre-shared key (PSK) and EAP enabled Wireless LANs.

Additionally the Level 1 MINT area-id will be modified so that the AP6522 client bridges will not be adopted and managed or seen by other WiNG 5 devices in the system. This ensures that any AP6522 client bridges accidently connected to the wired network will not be re-configured as infrastructure Access Points.

The following demonstrates how to modify the default AP6522 Profile following recommended best practices for standalone client bridge deployments:

1	Access the default AP6522 Profile configuration context:

```
AP6522-CB1(config)# profile ap6522 default-ap6522
```

2	Modify the Level 1 MINT area id (any value other than 1):

```
AP6522-CB1(config-profile-default-ap6522)# mint level 1 area-id 65535
```

3	Assign DNS Name Server, Domain Name and NTP server:

```
AP6522-CB1(config-profile-default-ap6522)# ip name-server 192.168.10.6
AP6522-CB1(config-profile-default-ap6522)# tmelabs.local
AP6522-CB1(config-profile-default-ap6522)# ntp server 192.168.10.1
```

4	Disable Radio 1:

```
AP6522-CB1(config-profile-default-ap6522)# interface radio 1
AP6522-CB1(config-profile-default-ap6522-if-radio1)# shutdown
AP6522-CB1(config-profile-default-ap6522-if-radio1)# exit
```

5A	Configure Radio 2 as a client bridge radio (PSK Example):

```
AP6522-CB1(config-profile-default-ap6522)# interface radio 2
AP6522-CB1(config-profile-default-ap6522-if-radio2)# rf-mode bridge
AP6522-CB1(config-profile-default-ap6522-if-radio2)# bridge ssid TMELABS-PSK
AP6522-CB1(config-profile-default-ap6522-if-radio2)# bridge authentication-type none
AP6522-CB1(config-profile-default-ap6522-if-radio2)# bridge encryption-type ccmp
AP6522-CB1(config-profile-default-ap6522-if-radio2)# bridge wpa-wpa2 psk hellomoto
AP6522-CB1(config-profile-default-ap6522-if-radio2)# exit
```

5B Configure Radio 2 as a client bridge radio (PEAP-MSCHAPv2 Example):

```
AP6522-CB1(config-profile-default-ap6522)# interface radio 2
AP6522-CB1(config-profile-default-ap6522-if-radio2)# rf-mode bridge
AP6522-CB1(config-profile-default-ap6522-if-radio2)# bridge ssid TMELABS-DOT1X
AP6522-CB1(config-profile-default-ap6522-if-radio2)# bridge authentication-type eap
AP6522-CB1(config-profile-default-ap6522-if-radio2)# bridge encryption-type ccmp
AP6522-CB1(config-profile-default-ap6522-if-radio2)# bridge eap username EAPUSER
AP6522-CB1(config-profile-default-ap6522-if-radio2)# bridge eap password hellomoto
AP6522-CB1(config-profile-default-ap6522-if-radio2)# exit
```

5C Configure Radio 2 as a client bridge radio (EAP-TLS Example):

```
AP6522-CB1(config-profile-default-ap6522)# interface radio 2
AP6522-CB1(config-profile-default-ap6522-if-radio2)# rf-mode bridge
AP6522-CB1(config-profile-default-ap6522-if-radio2)# bridge ssid TMELABS-DOT1X
AP6522-CB1(config-profile-default-ap6522-if-radio2)# bridge authentication-type eap
AP6522-CB1(config-profile-default-ap6522-if-radio2)# bridge encryption-type ccmp
AP6522-CB1(config-profile-default-ap6522-if-radio2)# bridge eap type tls
AP6522-CB1(config-profile-default-ap6522-if-radio2)# bridge eap username cb@tmelabs.local
AP6522-CB1(config-profile-default-ap6522-if-radio2)# bridge eap password hellomoto
AP6522-CB1(config-profile-default-ap6522-if-radio2)# exit
```

6 Exit then Commit and Write the Changes:

```
AP6522-CB1(config-profile-default-ap6522)# exit
AP6522-CB1(config)# commit write
```

Resulting Configuration Changes:

```
!
! PSK Example
!
profile ap6522 default-ap6522
 mint level 1 area-id 65535
 ip name-server 192.168.10.6
 ip domain-name tmelabs.local
 !
 ! Configuration Removed for Brevity
 !
 interface radio1
  shutdown
 interface radio2
  rf-mode bridge
  bridge ssid TMELABS-PSK
  bridge encryption-type ccmp
  bridge wpa-wpa2 psk 0 hellomoto
 interface ge1
  ip dhcp trust
  qos trust dscp
  qos trust 802.1p
 interface vlan 1
  ip address dhcp
  ip address zeroconf secondary
  ip dhcp client request options all
 interface pppoe1
 use firewall-policy default
 ntp server 192.168.10.1
 service pm sys-restart
 router ospf
!
```

```
!
! PEAP-MSCHAPv2 Example
!
profile ap6522 default-ap6522
 mint level 1 area-id 65535
 ip name-server 192.168.10.6
 ip domain-name tmelabs.local
 !
 ! Configuration Removed for Brevity
 !
 interface radio1
  shutdown
 interface radio2
  rf-mode bridge
  bridge ssid TMELABS-DOT1X
  bridge encryption-type ccmp
  bridge authentication-type eap
  bridge eap username eapuser1
  bridge eap password 0 hellomoto
 interface ge1
  ip dhcp trust
  qos trust dscp
  qos trust 802.1p
 interface vlan 1
  ip address dhcp
  ip address zeroconf secondary
  ip dhcp client request options all
 interface pppoe1
 use firewall-policy default
 ntp server 192.168.10.1
 service pm sys-restart
 router ospf
!
```

```
!
! EAP-TLS Example
!
profile ap6522 default-ap6522
 mint level 1 area-id 65535
 ip name-server 192.168.10.6
 ip domain-name tmelabs.local
 !
 ! Configuration Removed for Brevity
 !
 interface radio1
  shutdown
 interface radio2
  rf-mode bridge
  bridge ssid TMELABS-DOT1X
  bridge encryption-type ccmp
  bridge authentication-type eap
  bridge eap username cb@tmelabs.local
  bridge eap password 0 hellomoto
  bridge eap type tls
 interface ge1
  ip dhcp trust
  qos trust dscp
  qos trust 802.1p
 interface vlan 1
  ip address dhcp
  ip address zeroconf secondary
  ip dhcp client request options all
 interface pppoe1
 use firewall-policy default
 ntp server 192.168.10.1
 service pm sys-restart
 router ospf
!
```

2.1.5 Device Overrides

Each AP6522 client bridge will include device specific configuration parameters which are defined as overrides. Device specific parameters which can be defined include Hostnames for identification, area and floor assignments for Web-UI tree placement and static network addressing.

2.1.5.1 Hostname

By default each AP6522 client bridge is assigned a hostname which includes the device model and last three octets of the MAC address (<model>-<aabbcc>). As a general best practice it is recommended that you modify the default hostname for each deployed AP6522 Client Bridge for device identification.

The following demonstrates how to modify the hostname of an AP6522 client bridge for standalone client bridge deployments:

1 Access the Device configuration context:

```
ap6522-E91438(config)# ap6522 B4-C7-99-E9-14-38
```

2 Define a Hostname:

```
ap6522-E91438(config-device-B4-C7-99-E9-14-38)# hostname AP6522-CB1
```

3 Exit then Commit and Write the Changes:

```
ap6522-E91438(config-device-B4-C7-99-E9-14-38)# exit
ap6522-E91438(config)# commit write
```

Resulting Configuration Changes:

```
!
ap6522 B4-C7-99-E9-14-38
 use profile default-ap6522
 use rf-domain default
 hostname AP6522-CB1
!
```

*For standalone client bridge deployments you can directly access the Device configuration context by typing **self**.*

2.1.5.2 Static IPv4 Addressing

By default each AP6522 client bridge is configured to dynamically obtain IPv4 addressing from a DHCP server on VLAN 1. For certain deployments it may be desirable to define static IPv4 addressing which is defined along with a default gateway as a device override.

The following demonstrates how to define a static IPv4 address and default gateway on a AP6522 client bridge for standalone client bridge deployments:

1 Access the Device configuration context:

```
AP6522-CB1(config)# ap6522 B4-C7-99-E9-14-38
```

2 Define a Switched Virtual Interface (SVI) for VLAN 1 for Management:

```
AP6522-CB1(config-device-B4-C7-99-E9-14-38)# interface vlan 1
```

3 Define a IPv4 Address and Mask:

```
AP6522-CB1(config-device-B4-C7-99-E9-14-38-if-vlan1)# ip address 192.168.13.100/24
AP6522-CB1(config-device-B4-C7-99-E9-14-38-if-vlan1)# exit
```

4 Define a Default Gateway:

```
AP6522-CB1(config-device-B4-C7-99-E9-14-38)# ip default-gateway 192.168.13.1
```

5 Exit then Commit and Write the Changes:

```
AP6522-CB1(config-device-B4-C7-99-E9-14-38)# exit
AP6522-CB1(config)# commit write
```

Resulting Configuration Changes:

```
!
ap6522 B4-C7-99-E9-14-38
 use profile default-ap6522
 use rf-domain default
 hostname AP6522-CB1
 ip default-gateway 192.168.13.1
 interface vlan 1
  ip address 192.168.13.100/24
!
```

*For standalone client bridge deployments you can directly access the Device configuration context by typing **self**.*

2.1.6 Inline Password Encryption

Inline password encryption can be enabled to security encrypt and obfuscate any passwords, SNMP community strings and pre-shared keys (PSKs) stored in the running or startup configuration on an AP6522 client bridge. This prevents any sensitive information from being recovered if an AP6522 client bridge is stolen from a site.

As a general best practice is it recommended that inline password encryption be enabled for all AP6522 client bridge deployments. The feature is activated by first defining a password-encryption secret and then enabling the inline password encryption feature. Once enabled all passwords, SNMP community strings and PSKs will be encrypted using SHA256-AES256 encryption.

The following demonstrates how to enable inline password encryption on a AP6522 client bridge for standalone client bridge deployments:

1	**Access the Device configuration context:**

```
AP6522-CB1(config)# password-encryption secret 2 hellomoto
```

2	**Define a Hostname:**

```
AP6522-CB1(config)# inline-password-encryption
```

3	**Commit and Write the Changes:**

```
AP6522-CB1(config)# commit write
```

2.1.7 Staging

Standalone AP6522 client bridges require staging prior to being deployed in a production environment so that they can successfully connect and authenticate to the infrastructure Wireless LAN. For standalone deployments pre-staging involves applying the full configuration to each AP6522 client bridge which includes all the necessary parameters required to connect to the infrastructure Wireless LAN and manage the device.

2.1.7.1 Staging Configuration File

To simplify the deployment of standalone AP6522 client bridges it is recommended that you build a staging-config file which can be installed on each AP6522 client bridge. The staging-config file can be built on a single AP6522 client bridge and then exported so that it can be modified prior to deployment. For most standalone deployments the only unique value in the final configuration will be the hostname which can be defined on each individual AP6522 client bridge as part of the staging or deployment process.

When building a staging-config file it is important to update the device configuration section so that the template can be installed on multiple AP6522 client bridges. By default the device configuration section will include ***ap6522 <mac-address>*** which locks the configuration to a specific device. Changing the configuration from ***ap6522 <mac-address>*** to ***self*** allows the staging-config file to be installed on any AP6522 client bridge.

Device Configuration Example:

```
!
ap6522 B4-C7-99-E9-14-38
 use profile default-ap6522
 use rf-domain default
 hostname AP6522-CB1
!
```

➔

```
!
self
 use profile default-ap6522
 use rf-domain default
 hostname <NOT-DEFINED>
!
```

For convince example staging-config files are provided in the Appendix section of this guide.

If standalone AP6522 client bridges are deployed across multiple regions, a staging-config file will need to be built for each region with the appropriate country code defined in the default RF Domain.

2.1.7.2 Command Line Interface (CLI)

For small deployments the staging-config can be pasted directly into the command-line interface on each individual AP6522 client bridge. The staging-config can either be applied in the CLI using the serial console port or over Ethernet using an SSHv2 session. In both cases a direct CLI session is established to the target AP6522 client bridge and the staging-config is pasted directly into the CLI.

By default each AP6522 client bridge is enabled for DHCP and will attempt to obtain IPv4 addressing over VLAN 1 from its Ge1 port. If DHCP services are available an SSHv2 session can be established to the IPv4 address assigned from the DHCP server. If DHCP services are not available you can alternatively establish an SSHv2 connection using the AP6522 client bridges zero-configuration IPv4 address (RFC 3927).

The zero-configuration IP address assigned to each AP6522 client bridge uses a 169.254.0.0/16 prefix and the host portion of the address is the decimal equivalent of the last two octets of the AP6522 client bridges base MAC address. For example an AP6522 client bridge with the base MAC address B4-C7-99-E9-15-0C will have the zero-configuration IPv4 address 169.254.21.12/16.

*The default baud rate for the serial console port on the AP6522 client bridges is **115,200** baud.*

For management purposes and identification it is recommended that you assign a hostname to the AP6522 client bridges once the configuration template has been applied.

2.1.7.3 AutoInstall

For larger deployments the staging-config can be applied automatically using AutoInstall by connecting each AP6522 client bridge to a wired staging network. By default each AP6522 client bridge will automatically obtain IPv4 addressing on VLAN 1 when connected to a wired network. The DHCP server can be configured to supply Zebra vendor specific DHCP options which define the FTP or TFTP server parameters and the configuration filename and path.

AutoInstall is enabled by default in the default AP6522 Profile from the factory. When enabled the AP6522 client bridges will automatically connect to the FTP / TFTP server and download the template configuration. Once the staging-config is applied, the AP6522 client bridge will associate and authenticate to the defined Infrastructure Wireless LAN and can be deployed into the production environment.

Please refer to the WiNG 5 DHCP How-To Guide for a detailed overview of the available Zebra vendor-specific DHCP options in addition to DHCP server configuration examples.

It is strongly recommended that the AutoInstall be performed in an isolated staging environment. Once the configuration has been applied and radio 2 connects to the infrastructure Wireless LAN, a network loop can be formed between the infrastructure Wireless LAN and staging network if they reside on the same VLAN.

For management purposes and identification it is recommended that you assign a hostname to the AP6522 client bridges once the configuration template has been applied.

2.2 Centrally Managed Client Bridged Configuration

The following section provides an overview of the configuration steps required to deploy AP6522 Access Points as a centrally managed client bridges. Each AP6522 client bridge will be adopted and managed by a dedicated cluster of Centralized Controllers in the data center. The client bridges are deployed and managed in the same manner as Infrastructure Access Points in a ONEVIEW system.

All configuration will be performed directly on the Centralized Controllers and will be inherited using the command line interface (CLI) using the following Policies, Profile and RF Domain:

1. Default Firewall Policy
2. User Defined Management Policy
3. User Defined RF Domain
4. User Defined Profile

All the configuration will be performed on the Centralized Controllers and will be inherited by the AP6522 client bridges upon adoption and configuration.

2.2.1 Default Firewall Policy

Firewall Policies determine which firewall services are enabled on each WiNG 5 device. By default each WiNG 5 device is assigned the default Firewall Policy which is automatically mapped to default and user defined Profiles.

As the AP6522 will be deployed as a client bridge and not an infrastructure Access Point, all firewall services will be disabled. Stateful packet inspection and policy enforcement will be primarily provided by the Wireless LAN infrastructure and not the AP6522 client bridges. It is possible however to define and assign IP or MAC Access Control Lists (ACLs) to the Ge1 port if required.

The following demonstrates how to modify the default Firewall Policy on the Centralized Controllers following recommended best practices centrally managed client bridge deployments:

1	**Access the default Firewall Policy configuration context:**

```
CBMGR-ACTIVE(config)# firewall-policy default
```

2	**Disable DoS detection:**

```
CBMGR-ACTIVE(config-fw-policy-default)# no ip dos
```

3	**Disable IP MAC Conflict detection:**

```
CBMGR-ACTIVE(config-fw-policy-default)# no ip-mac conflict
CBMGR-ACTIVE(config-fw-policy-default)# no ip-mac routing conflict
```

4	**Disable Layer 2 Stateful Packet Inspection:**

```
CBMGR-ACTIVE(config-fw-policy-default)# no stateful-packet-inspection-l2
```

5 Disable the Layer 3 Firewall:

```
CBMGR-ACTIVE(config-fw-policy-default)# no firewall enable
```

6 Exit then Commit and Write the Changes:

```
CBMGR-ACTIVE(config-fw-policy-default)# exit
CBMGR-ACTIVE(config)# commit write
```

Resulting Configuration Changes:

```
!
firewall-policy default
 no ip dos smurf
 no ip dos twinge
 no ip dos invalid-protocol
 no ip dos router-advt
 no ip dos router-solicit
 no ip dos option-route
 no ip dos ascend
 no ip dos chargen
 no ip dos fraggle
 no ip dos snork
 no ip dos ftp-bounce
 no ip dos tcp-intercept
 no ip dos broadcast-multicast-icmp
 no ip dos land
 no ip dos tcp-xmas-scan
 no ip dos tcp-null-scan
 no ip dos winnuke
 no ip dos tcp-fin-scan
 no ip dos udp-short-hdr
 no ip dos tcp-post-syn
 no ip dos tcphdrfrag
 no ip dos ip-ttl-zero
 no ip dos ipspoof
 no ip dos tcp-bad-sequence
 no ip dos tcp-sequence-past-window
 no ip-mac conflict
 no ip-mac routing conflict
 no firewall enable
 no stateful-packet-inspection-l2
!
```

2.2.2 User Defined Management Policy

Management Policies determine which management services and administrative user accounts are enabled on each WiNG 5 device. Management Policies also determine how the administrative user accounts are authenticated and authorized. By default each WiNG 5 device is assigned the default Management Policy which is automatically mapped to default and user defined Profiles.

As a general best practice it is recommended that you create a user defined Management Policy with all management services disabled except SSHv2. The user defined Management Policy will be mapped to each defined Profile assigned to the AP6522 client bridges. Remote management access should also be restricted to specific hosts or subnets.

The following demonstrates how to create a user defined Management Policy following recommended best practices for centrally managed client bridge deployments:

1	**Access the default Management Policy configuration context:**

```
CBMGR-ACTIVE(config)# management-policy CLIENT-BRIDGES
```

2	**Define an administrative Username and Password (Mandatory):**

```
CBMGR-ACTIVE(config-management-policy-CLIENT-BRIDGES)# user admin password hellomoto role superuser access all
```

3	**Disable un-necessary management services:**

```
CBMGR-ACTIVE(config-management-policy-CLIENT-BRIDGES)# no http server
CBMGR-ACTIVE(config-management-policy-CLIENT-BRIDGES)# no snmp-server manager all
```

4	**Enable SSHv2:**

```
CBMGR-ACTIVE(config-management-policy-CLIENT-BRIDGES)# ssh
```

5	**Restrict remote management access to a specific Host(s) or Subnet(s):**

```
CBMGR-ACTIVE(config-management-policy- CLIENT-BRIDGES)# restrict-access subnet 192.168.10.0/24
```

6	**Exit then Commit and Write the Changes:**

```
CBMGR-ACTIVE(config-management-policy- CLIENT-BRIDGES)# exit
CBMGR-ACTIVE(config)# commit write
```

Resulting Configuration Changes:

```
!
management-policy CLIENT-BRIDGES
 no http server
 ssh
 user admin password 1 <obfuscated-password> role superuser access all
 no snmp-server manager v3
 restrict-access subnet 192.168.10.0/24
!
```

2.2.3 Default RF Domain

RF Domains determine regulatory a regional operating parameters for the AP6522 client bridges and in managed deployments are also used for organization and visualization. By default each WiNG 5 device is assigned to a default RF Domain which is automatically mapped to the devices configuration.

For centrally managed deployments one user defined RF Domain is required for each remote site where the AP6522 client bridges are deployed. Each user defined RF Domain is configured as Controller Managed so that the Active Centralized Controller is the elected RF Domain Manager for each RF Domain. As no Control-VLAN will be established between the AP6522 client bridges, the Active Centralized Controller must assume the RFDM role.

Each user defined RF Domain requires that an ISO 3166 Country Code to be defined and as a general best practice it is also recommended that each RF Domain includes a Location, Contact and TimeZone to simplify the identification and management of the client bridges.

The following demonstrates how to create a user defined RF Domain following recommended best practices for centrally managed client bridge deployments:

1 Create a user defined RF Domain:

```
CBMGR-ACTIVE(config)# rf-domain STORE201
```

2 Define a Location, Contact, Country Code and TimeZone:

```
CBMGR-ACTIVE(config-rf-domain-STORE201)# location “Johnson City TN”
CBMGR-ACTIVE(config-rf-domain-STORE201)# contact “kmarshall@Zebrasolutions.com”
CBMGR-ACTIVE(config-rf-domain-STORE201)# country-code us
CBMGR-ACTIVE(config-rf-domain-STORE201)# timezone EST5EDT
```

3 Configure the RF Domain as Controller Managed (Mandatory):

```
CBMGR-ACTIVE(config-rf-domain-STORE201)# controller-managed
```

4 Exit then Commit and Write the Changes:

```
CBMGR-ACTIVE(config-rf-domain-STORE201)# exit
CBMGR-ACTIVE(config)# commit write
```

Resulting Configuration Changes:

```
!
rf-domain STORE201
 location “Johnson City TN”
 contact kmarshall@Zebrasolutions.com
 timezone EST5EDT
 country-code us
 controller-managed
!
```

2.2.4 User Defined AP6522 Profile

Profiles assign common configuration parameters to groups of managed WiNG5 devices and are model specific. By default each WiNG 5 device is assigned to a default Profile which is automatically mapped to the devices configuration.

For centrally managed deployments one or more user defined AP6522 Profiles are defined for the AP6522 client bridges. Each user defined Profile will be modified to assign DNS and NTP parameters in addition to configuring radio 2 as a client bridge radio to connect to a Wireless LAN serviced by the infrastructure Access Points. Examples will be provided to configure the client bridge radio to connect to pre-shared key (PSK) and EAP enabled Wireless LANs.

Additionally the Level 1 MINT area-id will be modified so that the AP6522 client bridges will not be adopted and managed or seen by other WiNG 5 devices in the system. This ensures that any AP6522 client bridges accidently connected to the wired network will not be re-configured as infrastructure Access Points.

The following demonstrates how to create a user defined AP6522 Profile following recommended best practices for centrally managed client bridge deployments:

1 Create a user defined AP6522 Profile:

```
CBMGR-ACTIVE(config)# profile ap6522 CB-AP6522
```

2 Modify the Level 1 MINT area id (any value other than 1):

```
CBMGR-ACTIVE(config-profile-CB-AP6522)# mint level 1 area-id 65535
```

3 Assign DNS Name Server, Domain Name and NTP server:

```
CBMGR-ACTIVE(config-profile-CB-AP6522)# ip name-server 192.168.10.6
CBMGR-ACTIVE(config-profile-CB-AP6522)# tmelabs.local
CBMGR-ACTIVE(config-profile-CB-AP6522)# ntp server 192.168.10.1
```

4 Map the user defined Management Policy:

```
CBMGR-ACTIVE(config-profile-CB-AP6522)# use management-policy CLIENT-BRIDGES
```

5 Define a Switched Virtual Interface (SVI) for VLAN 1 enabled for DHCP for Management &

	Adoption:

```
CBMGR-ACTIVE(config-profile-CB-AP6522)# interface vlan 1
CBMGR-ACTIVE(config-profile-CB-AP6522-if-vlan1)# ip address dhcp
CBMGR-ACTIVE(config-profile-CB-AP6522-if-vlan1)# ip dhcp client request options all
CBMGR-ACTIVE(config-profile-CB-AP6522-if-vlan1)# exit
```

6	Disable Radio 1:

```
CBMGR-ACTIVE(config-profile-CB-AP6522)# interface radio 1
CBMGR-ACTIVE(config-profile-CB-AP6522-if-radio1)# shutdown
CBMGR-ACTIVE(config-profile-CB-AP6522-if-radio1)# exit
```

7A	Configure Radio 2 as a client bridge radio (PSK Example):

```
CBMGR-ACTIVE(config-profile-CB-AP6522)# interface radio 2
CBMGR-ACTIVE(config-profile-CB-AP6522-if-radio2)# rf-mode bridge
CBMGR-ACTIVE(config-profile-CB-AP6522-if-radio2)# bridge ssid TMELABS-PSK
CBMGR-ACTIVE(config-profile-CB-AP6522-if-radio2)# bridge authentication-type none
CBMGR-ACTIVE(config-profile-CB-AP6522-if-radio2)# bridge encryption-type ccmp
CBMGR-ACTIVE(config-profile-CB-AP6522-if-radio2)# bridge wpa-wpa2 psk hellomoto
CBMGR-ACTIVE(config-profile-CB-AP6522-if-radio2)# exit
```

7B	Configure Radio 2 as a client bridge radio (PEAP-MSCHAPv2 Example):

```
CBMGR-ACTIVE(config-profile-CB-AP6522)# interface radio 2
CBMGR-ACTIVE(config-profile-CB-AP6522-if-radio2)# rf-mode bridge
CBMGR-ACTIVE(config-profile-CB-AP6522-if-radio2)# bridge ssid TMELABS-DOT1X
CBMGR-ACTIVE(config-profile-CB-AP6522-if-radio2)# bridge authentication-type eap
CBMGR-ACTIVE(config-profile-CB-AP6522-if-radio2)# bridge encryption-type ccmp
CBMGR-ACTIVE(config-profile-CB-AP6522-if-radio2)# bridge eap username EAPUSER
CBMGR-ACTIVE(config-profile-CB-AP6522-if-radio2)# bridge eap password hellomoto
CBMGR-ACTIVE(config-profile-CB-AP6522-if-radio2)# exit
```

7C	Configure Radio 2 as a client bridge radio (EAP-TLS Example):

```
CBMGR-ACTIVE(config-profile-CB-AP6522)# interface radio 2
CBMGR-ACTIVE(config-profile-CB-AP6522-if-radio2)# rf-mode bridge
CBMGR-ACTIVE(config-profile-CB-AP6522-if-radio2)# bridge ssid TMELABS-DOT1X
CBMGR-ACTIVE(config-profile-CB-AP6522-if-radio2)# bridge authentication-type eap
CBMGR-ACTIVE(config-profile-CB-AP6522-if-radio2)# bridge encryption-type ccmp
CBMGR-ACTIVE(config-profile-CB-AP6522-if-radio2)# bridge eap type tls
CBMGR-ACTIVE(config-profile-CB-AP6522-if-radio2)# bridge eap username cb@tmelabs.local
CBMGR-ACTIVE(config-profile-CB-AP6522-if-radio2)# bridge eap password hellomoto
CBMGR-ACTIVE(config-profile-CB-AP6522-if-radio2)# exit
```

8 Exit then Commit and Write the Changes:

```
CBMGR-ACTIVE(config-profile-CB-AP6522)# exit
CBMGR-ACTIVE(config)# commit write
```

Resulting Configuration Changes:

```
!
! PSK Example
!
profile ap6522 CB-AP6522
 mint level 1 area-id 65535
 ip name-server 192.168.10.6
 ip domain-name tmelabs.local
 !
 ! Configuration Removed for Brevity
 !
 interface radio1
  shutdown
 interface radio2
  rf-mode bridge
  bridge ssid TMELABS-PSK
  bridge encryption-type ccmp
  bridge wpa-wpa2 psk 0 hellomoto
 interface ge1
  ip dhcp trust
  qos trust dscp
  qos trust 802.1p
 interface vlan 1
  ip address dhcp
  ip dhcp client request options all
 interface pppoe1
 use management-policy CLIENT-BRIDGES
 use firewall-policy default
 ntp server 192.168.10.1
 service pm sys-restart
 router ospf
!
```

```
!
! PEAP-MSCHAPv2 Example
!
profile ap6522 CB-AP6522
 mint level 1 area-id 65535
 ip name-server 192.168.10.6
 ip domain-name tmelabs.local
 !
 ! Configuration Removed for Brevity
 !
 interface radio1
  shutdown
 interface radio2
  rf-mode bridge
  bridge ssid TMELABS-DOT1X
  bridge encryption-type ccmp
  bridge authentication-type eap
  bridge eap username eapuser1
  bridge eap password 0 hellomoto
 interface ge1
  ip dhcp trust
  qos trust dscp
  qos trust 802.1p
interface vlan 1
  ip address dhcp
  ip dhcp client request options all
 interface pppoe1
 use management-policy CLIENT-BRIDGES
 use firewall-policy default
 ntp server 192.168.10.1
 service pm sys-restart
 router ospf
!
```

```
!
! EAP-TLS Example
!
profile ap6522 CB-AP6522
 mint level 1 area-id 65535
 ip name-server 192.168.10.6
 ip domain-name tmelabs.local
 !
 ! Configuration Removed for Brevity
 !
 interface radio1
  shutdown
 interface radio2
  rf-mode bridge
  bridge ssid TMELABS-DOT1X
  bridge encryption-type ccmp
  bridge authentication-type eap
  bridge eap username cb@tmelabs.local
  bridge eap password 0 hellomoto
  bridge eap type tls
 interface ge1
  ip dhcp trust
  qos trust dscp
  qos trust 802.1p
interface vlan 1
  ip address dhcp
  ip dhcp client request options all
 interface pppoe1
 use management-policy CLIENT-BRIDGES
 use firewall-policy default
 ntp server 192.168.10.1
 service pm sys-restart
 router ospf
!
```

2.2.5 Inline Password Encryption

Inline password encryption can be enabled to security encrypt and obfuscate passwords, SNMP community strings and pre-shared keys (PSKs) stored in the running or startup configuration on both the Centralized Controllers and remote AP6522 client bridges.

As a general best practice is it recommended that inline password encryption be enabled for all AP6522 client bridge deployments. The feature is activated by first defining a password-encryption secret and then enabling the inline password encryption feature. Once enabled all passwords, SNMP community strings and PSKs will be encrypted using SHA256-AES256 encryption.

The following demonstrates how to enable inline password encryption on the Centralized Controllers for centrally managed deployments:

1	**Access the Device configuration context:**

```
CBMGR-ACTIVE(config)# password-encryption secret 2 hellomoto
```

2	**Define a Hostname:**

```
CBMGR-ACTIVE(config)# inline-password-encryption
```

3	**Commit and Write the Changes:**

```
CBMGR-ACTIVE(config)# commit write
```

2.2.6 Centralized Controller Discovery

For centrally managed deployments AP6522 client bridges can discover the Centralized Controllers and establish Level 2 MINT links using static Controller Host entries or DHCP options. The choice as to which discovery option to enable will depend on the environment in which the AP6522 client bridges are being deployed.

*If a centrally managed AP6522 client bridge is connected to a WiNG5 infrastructure Access Point that is tunneling to a Site Controller, you must enable the **mint tunnel-across-extended-vlan** parameter in both the Site Controller and infrastructure Access Point Profiles. By default WiNG 5 will prevent MINT packets from being tunneled over an extended VLAN which results in the remote AP6522 client bridges from being able to discover the Centralized Controllers and adopting.*

2.2.6.1 DHCP Option 191

AP6522 client bridges support dynamic layer 3 discovery using Dynamic Host Control Protocol (DHCP). DHCP can be used to assign network addressing in addition to sending Zebra vendor-specific DHCP option 191 which is used by the AP6522 client bridges to automatically discover the Centralized Controllers and adopt. DHCP option 191 is supplied as ASCII / string to the AP6522 client bridges in a DHCP offer and each string includes the IPv4 addresses of the Active and Standby Centralized Controllers, Pool and MINT routing level. The Active Controller IP address is assigned to Pool1 while the Standby Controller IP address is assigned to Pool2. The MINT routing level is set to 2.

Option 191 can be assigned directly to each DHCP pool assigning network addresses to the AP6522 client bridges or globally to all devices using option 60 vendor class. The AP6522 client bridges will identify themselves as ZebraAP.AP6522 in both DHCP discover and acknowledgement packets.

Option	Format	Example
191	ASCII / String	pool1=<active-controller-ip>;pool2=<standby-controller-ip>;level=2

Table 2.2.6.1 – DHCP Option 191 String Formatting

Please refer to the WiNG 5 DHCP How-To Guide for a detailed overview of the available Zebra vendor-specific DHCP options in addition to DHCP server configuration examples.

*To support the AP6522 client bridges, **DHCP Option 191** must be assigned to each DHCP Pool providing addresses to the AP6522 client bridges.*

2.2.6.2 Controller Hosts

AP6522 client bridges support static layer 3 discovery using Controller Hosts defined in the user defined AP6522 Profiles. Controller Hosts are typically used for deployments when AP6522 client bridges are assigned static network addressing or when DHCP options cannot be deployed.

Each Controller Host entry includes the IPv4 address of the Active or Standby Centralized Controller, Pool and MINT routing Level. The AP6522 Profile will include up to two Controller Host entries where the first entry defines the Active Controller IP address (Pool 1) while the second entry defines the Standby Controller IP address (Pool 2). The MINT routing level is set to 2 for both entries.

Command Syntax (Profile or Device)
controller host <controller-ip-address> [pool1 | pool2] level 2

Table 2.2.6.3 – Controller Host Syntax

1 Access the Client Bridge Profile configuration context:

```
CBMGR-ACTIVE(config)# profile ap6522 CB-AP6522
```

2 Define Controller Host entries for the Active and Standby Centralized Controllers:

```
CBMGR-ACTIVE(config-profile-CB-AP6522)# controller host 192.168.20.90 pool 1 level 2
CBMGR-ACTIVE(config-profile-CB-AP6522)# controller host 192.168.20.91 pool 2 level 2
```

3 Exit then Commit and Write the Changes:

```
CBMGR-ACTIVE(config-profile-CB-AP6522)# exit
CBMGR-ACTIVE(config)# commit write
```

Resulting Configuration Changes:

```
!
profile ap6522 CB-AP6522
 mint level 1 area-id 65535
 ip name-server 192.168.10.6
 !
 ! Configuration Removed for Brevity
 !
 controller host 192.168.20.90 pool 1 level 2
 controller host 192.168.20.91 pool 2 level 2
 service pm sys-restart
 router ospf
!
```

2.2.7 Auto-Provisioning

For centrally managed deployments each AP6522 client bridge is assigned a user defined Profile and RF Domain upon adoption. The user defined Profile and RF Domain are assigned to new AP6522 client bridges upon the initial adoption using an Auto Provisioning Policy that is mapped to the Centralized Controller Profile. The Auto-Provisioning Policy include adopt rules that determines the Profile and RF Domain to assign to each new AP6522 client bridge based on a defined match type and value.

2.2.7.1 IP Match Type

The IP match type can be used for centrally managed deployments where no hostnames are pre-defined on the AP6522 Client Bridges prior to deployment.

Each adopt rule assigns a single Profile and RF Domain to AP6522 client bridges based on the unique IPv4 subnet the client bridges are connected to. The remote Client bridges at each remote site connect to the Infrastructure Wireless LAN and are assigned a unique IPv4 address which is site specific. All AP6522 client bridges at a site will be mapped to the same Profile and RF Domain.

The following demonstrates how to create an Auto Provisioning Policy with IP match rules and map the Policy to the Centralized Controller Profile for centrally managed client bridge deployments:

1 Create a user defined Auto Provisioning Policy:

```
CBMGR-ACTIVE(config)# auto-provisioning-policy DATACENTER
```

2 Define IP match adoption rule for Profile and RF Domain assignment :

```
CBMGR-ACTIVE(config-auto-provisioning-policy-DATACENTER)# adopt ap6522 precedence 1 profile
CB-AP6522 rf-domain STORE201 ip 192.168.21.0/24
CBMGR-ACTIVE(config-auto-provisioning-policy-DATACENTER)# exit
```

3 Map the Auto Provisioning Policy to the Centralized Controller Profile:

```
CBMGR-ACTIVE(config)# profile nx9000 DATACENTER-NX9000
CBMGR-ACTIVE(config-profile-NX9000-DATACENTER)# use auto-provisioning-policy DAT
ACENTER
```

4 Exit then Commit and Write the Changes:

```
CBMGR-ACTIVE(config-profile-NX9000-DATACENTER)# exit
CBMGR-ACTIVE(config)# commit write
```

When the IP match type is used, one adopt rule is required per remote site.

2.2.7.2 FQDN Wildcards

The FQDN wildcard match type can be used for centrally managed deployments when hostnames are pre-defined on the AP6522 Client Bridges prior to deployment. Depending on the deployment FQDN wildcard rules can be used to assign both user defined Profiles and RF Domains when SSIDs, passwords or pre-shared keys differ between brands. FQDN wildcards may also be used to assign user defined RF Domains along with a common user defined Profile when SSIDs, passwords or pre-shared keys are common between sites.

2.2.7.2.1 RF Domain Assignment

An FQDN wildcard allows a common Profile and unique RF Domain to be assigned to a new AP6522 client bridges based on string values contained within a pre-staged hostname of the adopting device. For FQDN wildcards to function, the pre-staged hostname must include a site identifier at a fixed position which can be used by the Auto-Provisioning policy to determine and assign the correct user defined RF Domain.

For example AP6522 client bridges can be pre-provisioned with hostnames using a STXXXMOTCBYY format where XXX defines the site identifier and YY defines the device identifier. Using a single FQDN wildcard rule the 3rd → 5th characters (XXX) can be matched to assign an RF Domain named STOREXXX. The pre-staged hostname ST201MOTCB01 in this case could result in the user defined RF Domain named STORE201 being assigned to the device. A common Profile would be assigned to all the AP6522 client bridges using a second rule in the Policy with an Any match type.

The following demonstrates how to create an Auto Provisioning Policy with FQDN wildcard rule for RF Domain assignment and map the Policy to the Centralized Controller Profile for centrally managed client bridge deployments:

1 Create a user defined Auto Provisioning Policy:

```
CBMGR-ACTIVE(config)# auto-provisioning-policy DATACENTER
```

2 Define a IP match adoption rule for Profile assignment:

```
CBMGR-ACTIVE(config-auto-provisioning-policy-DATACENTER)# adopt ap6522 precedence 1 profile CB-AP6522 any
```

3 Define a FQDN wildcard adoption rule for RF Domain assignment:

```
CBMGR-ACTIVE(config-auto-provisioning-policy-DATACENTER)# adopt ap6522 precedence 2 rf-domain STORE$FQDN[3:5] any
```

4 Map the Auto Provisioning Policy to the Centralized Controller Profile:

```
CBMGR-ACTIVE(config)# profile nx9000 DATACENTER-NX9000
CBMGR-ACTIVE(config-profile-NX9000-DATACENTER)# use auto-provisioning-policy DATACENTER
```

5 Exit then Commit and Write the Changes:

```
CBMGR-ACTIVE(config-profile-NX9000-DATACENTER)# exit
CBMGR-ACTIVE(config)# commit write
```

2.2.7.2.2 Profile and RF Domain Assignment

An FQDN wildcard allows a unique Profile and RF Domain to be assigned to a new AP6522 client bridges based on string values contained within a pre-staged hostname of the adopting device. For FQDN wildcards to function, the pre-staged hostname must include a brand and site identifier at a fixed position which can be used by the Auto-Provisioning policy to determine and assign the correct user defined Profile and RF Domain.

For example the AP6522 client bridges can be pre-provisioned with the hostname XXSTYYYMOTCBZZ where XX defines the brand, YYY defines the site identifier and ZZ defines the device identifier. Using one FQDN wildcard rule the 1st → 2nd characters (XX) can be matched to assign a Profile named CB-XX-AP6522. The pre-staged hostname WMST201CB01 in this case could result in the user defined Profile named CB-WM-AP6522 being assigned to the device.

A second FQDN wildcard rule would be defined to determine the user defined RF Domain assignment. The 5th → 7th characters (YYY) can be matched to assign a RF Domain named STOREYYY. The pre-staged hostname WMST201CB01 in this case could result in the user defined RF Domain named STORE201 being assigned to the device.

The following demonstrates how to create an Auto Provisioning Policy with FQDN wildcard rules for both Profile and RF Domain assignment and map the Policy to the Centralized Controller Profile for Centrally Managed client bridge deployments:

1 Create a user defined Auto Provisioning Policy:

```
CBMGR-ACTIVE(config)# auto-provisioning-policy DATACENTER
```

2 Define a FQDN wildcard adoption rule for Profile assignment:

```
CBMGR-ACTIVE(config-auto-provisioning-policy-DATACENTER)# adopt ap6522 precedence 1 profile
CB-$FQDN[1:2]-AP6522 any
```

3 Define a FQDN wildcard adoption rule for RF Domain assignment:

```
CBMGR-ACTIVE(config-auto-provisioning-policy-DATACENTER)# adopt ap6522 precedence 2 rf-domain
STORE$FQDN[3:5] any
```

4 Map the Auto Provisioning Policy to the Centralized Controller Profile:

```
CBMGR-ACTIVE(config)# profile nx9000 DATACENTER-NX9000
CBMGR-ACTIVE(config-profile-NX9000-DATACENTER)# use auto-provisioning-policy DATACENTER
```

5 Exit then Commit and Write the Changes:

```
CBMGR-ACTIVE(config-profile-NX9000-DATACENTER)# exit
CBMGR-ACTIVE(config)# commit write
```

2.2.8 Staging

Centrally managed AP6522 client bridges require pre-configuration prior to being deployed in a production environment so that they can successfully connect and authenticate to the infrastructure Wireless LAN and discover the Centralized Controllers. As centrally managed deployments tend to be larger than standalone deployments, plug-n-play staging is required.

The goal of staging AP6522 client bridges for centrally managed deployments is to get an AP6522 client bridge to a point where it associates and authenticates to the infrastructure Wireless LAN. This is achieved by automatically installing a staging-config onto each AP6522 client bridge either using AutoInstall or by adoption to a staging controller such as an RFS4000 or RFS6000. Once an AP6522 client bridge has received its staging-config it can be deployed into a production environment.

2.2.8.1 AutoInstall

AutoInstall provides a plug-n-play mechanism to automatically upgrade firmware images and/or install configuration files on AP6522 client bridges from a TFTP or FTP server using Zebra vendor-specific DHCP options. New AP6522 client bridges can be connected to a pre-staging network and using DHCP options automatically download and upgrade their firmware to the correct release in addition to downloading and installing a staging-config. Once completed the AP6522 client bridges can be deployed into a production environment:

Step 1	A new AP6522 client bridge is connected to the staging network. The AP6522 client bridge will obtain IPv4 addressing from a DHCP server along with Zebra vendor-specific DHCP options.
Step 2	If required the AP6522 client bridge will download and install the correct firmware image file from the FTP/TFTP server and reboot.
Step 3	The AP6522 client bridge will download and install the staging-config file from the FTP/TFTP server. If the infrastructure Wireless LAN is reachable it will associate and authenticate.
Step 4	Optionally assign a hostname to the AP6522 client bridge if FQDN wildcard auto-provisioning is employed.
Step 5	The AP6522 client bridge is now fully pre-staged and can be deployed in a production environment.

Table 2.2.8.1 – AutoInstall Staging Process

Once the staging-config has been applied, the AP6522 client bridges can now be deployed into a production environment. When an AP6522 client bridge is powered-on it will associate and authenticate to the infrastructure Wireless LAN, obtain IPv4 addressing and using DHCP option 191 or Controller Host entries to discover the Active Centralized Controller. Using auto-provisioning the Active Centralized Controller will assign the correct user defined Profile and RF Domain to the new AP6522 client bridge, add the new AP6522 client bridge to the master-config and apply the new configuration to the AP6522 client bridge.

Please refer to the WiNG 5 DHCP How-To Guide for a detailed overview of the available Zebra vendor-specific DHCP options in addition to DHCP server configuration examples.

For convince example staging-config files are provided in the Appendix section of this guide.

When pre-staging multiple AP6522 client bridges using AutoInstall, to prevent loops it is strongly recommended that the Ge1 port in the staging-config be disabled. The Ge1 port can be re-activated once the final configuration has been applied from the Active Centralized Controller.

For management purposes and identification it is recommended that you assign a hostname to the AP6522 client bridges once the staging-config has been applied.

2.2.8.2 Staging Controller

An alternative plug-n-play option to AutoInstall is to pre-stage the AP6522 client bridges on a Staging Controller such as a RFS4000 or RFS6000. For ease of deployment new AP6522 client bridges can be adopted at Layer 2 allowing the firmware to be upgraded and the pre-staged configuration to be applied. The AP6522 client bridges can either be directly connected to a Ge port on the Staging Controller or connected to common VLAN via a Layer 2 switch. Once upgraded and adopted the AP6522 client bridges can be deployed into a production environment:

Step 1	A new AP6522 client bridge is connected to the Staging Controller or Layer 2 switch. The AP6522 client bridge will discover the Staging Controller at Layer 2 and request adoption.
Step 2	If a version-mismatch is detected, the Staging Controller will upgrade the AP6522 client bridge to correct firmware version and reboot it.
Step 3	Upon re-adoption the Staging Controller will apply the configuration.
Step 4	Using the CLI or Web-UI optionally assign a hostname to the AP6522 client bridge if FQDN wildcard auto-provisioning is employed.
Step 5	The AP6522 client bridge is now fully pre-staged and can be deployed in a production environment.

Table 2.2.8.2 – Staging Controller Staging Process

Once the staging-config has been applied, the AP6522 client bridges can now be deployed into a production environment. When an AP6522 client bridge is powered-on it will associate and authenticate to the infrastructure Wireless LAN, obtain IPv4 addressing and using DHCP option 191 or Controller Host entries to discover the Active Centralized Controller. Using auto-provisioning the Active Centralized Controller will assign the correct user defined Profile and RF Domain to the new AP6522 client bridge, add the new AP6522 client bridge to the master-config and apply the new configuration to the AP6522 client bridge.

For convince example staging-config files are provided in the Appendix section of this guide.

When pre-staging multiple AP6522 client bridges using a Staging Controller, to prevent loops it is strongly recommended that the Ge1 port in the staging-config be disabled. The Ge1 port can be re-activated once the final configuration has been applied from the Active Centralized Controller.

For management purposes and identification it is recommended that you assign a hostname to the AP6522 client bridges once the staging-config has been applied.

2.3 EAP Credentials

2.3.1 PEAP MSCHAPv2

Protected Extensible Authentication Protocol (PEAP) is a member of the family of Extensible Authentication Protocol (EAP) protocols. PEAP uses Transport Layer Security (TLS) to create an encrypted channel between the AP6522 client bridge and the backend RADIUS server. PEAP does not specify an authentication method, but provides additional security for other EAP authentication protocols such as EAP-MSCHAPv2 which operates through the TLS encrypted channel provided by PEAP.

PEAP uses usernames and passwords for authentication and does not require any X.509 client-side certificates to be deployed. While CA root certificates are typically used to validate the RADIUS server, this is not currently implemented on the AP6522 client bridges.

2.3.1.1 Common vs. Unique Credentials

The AP6522 client bridges are configured with a unique or common username and password depending on your security requirements. Both options are supported for standalone and centrally managed deployments and the deployment method selected will depend on how you wish to balance your security and administrative requirements.

Unique credentials greatly increases administrative / management overhead of the AP6522 client bridges but provides additional security. When unique credentials are deployed each AP6522 client bridge requires a unique username and password to be defined in the back-end user directory in addition to unique ***bridge eap username*** and ***bridge eap password*** parameters that are defined as Device overrides.

The main advantage of deploying unique credentials is that it is very easy for an administrator to disable a single user account if an individual AP6522 client bridge is stolen or compromised. A second advantage is that each AP6522 client bridge can be assigned a unique username to simplify identification on the infrastructure Wireless LAN.

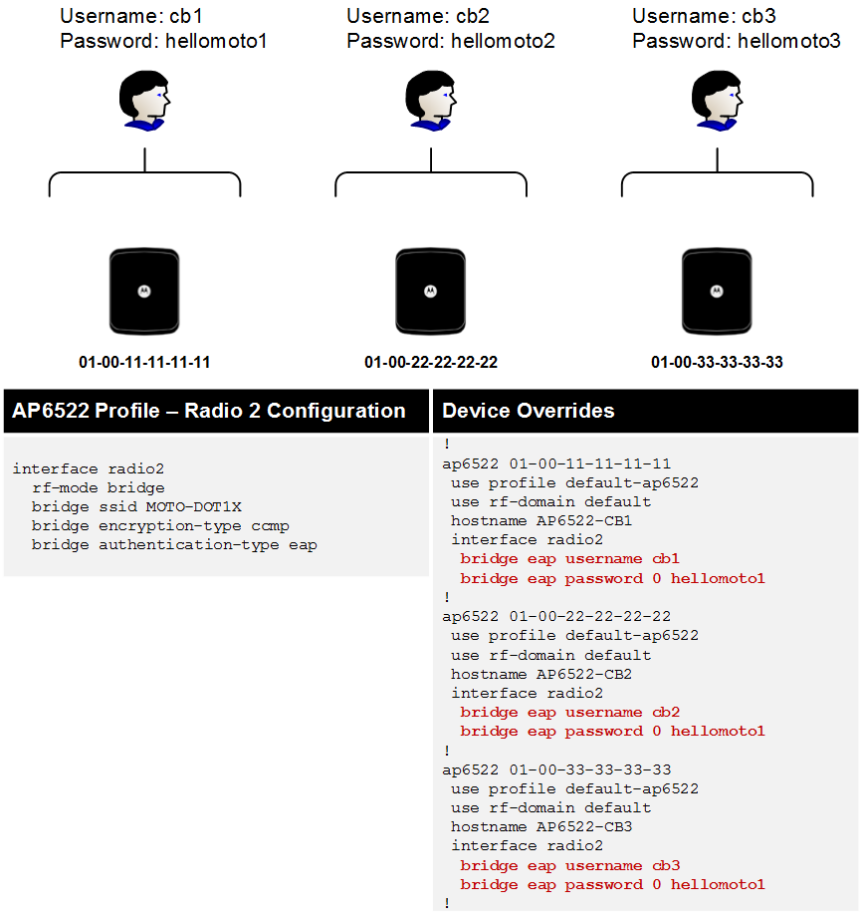


Figure 2.3.1.1-1 Unique PEAP Credentials Configuration Example

Common credentials greatly reduces the administrative / management overhead of the AP6522 client bridges as a single username and password is used to authenticate all the AP6522 client bridges in the system. Administrators define a single service account in the user directory or RADIUS server with no password expiration. As a single username and password is defined, the ***bridge eap username*** and ***bridge eap password*** parameters can be assigned to the AP6522 Profile and no Device overrides are required.

The main dis-advantage of deploying common credentials is that an administrator cannot easily disable the user account if an individual AP6522 client bridge is stolen or compromised. If the credentials are compromised the usernames and passwords on all the AP6522 client bridges must be updated before the compromised user account can be disabled. Additionally using a single username may make device identification more challenging on the infrastructure Wireless LAN.

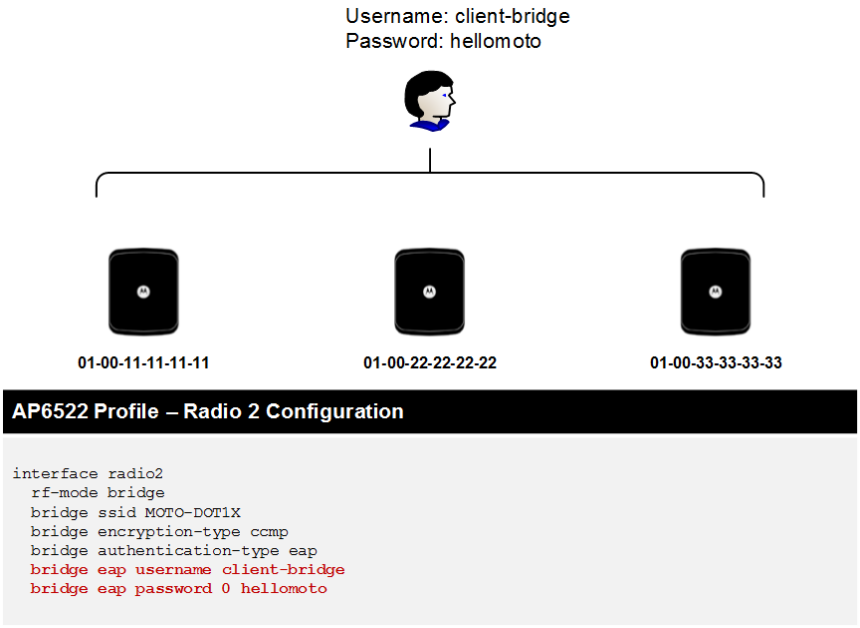


Figure 2.3.1.1-2 Common PEAP Credentials Configuration Example

☑	*It is strongly recommended that you enable **Inline Password Encryption** to protect the account password. This provides an additional security level in the event that an AP6522 client bridge is stolen or compromised.*
☑	*For centrally managed deployments requiring unique usernames and passwords, it is recommended that a common username and password be used for staging which can then be changed using Device overrides once the AP6522 client bridges have been adopted by the Active Centralized Controller.*

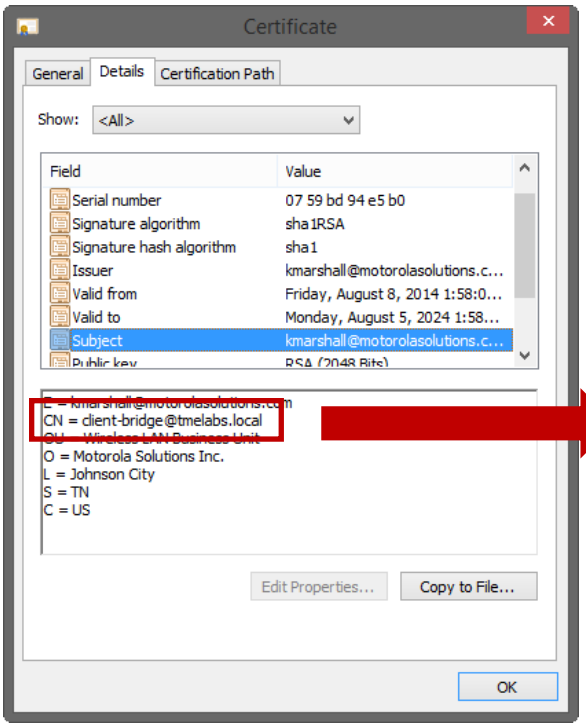
2.3.2 EAP-TLS

AP6522 client bridges can authenticate to the infrastructure Wireless LAN using EAP-Transport Layer Security (TLS). EAP-TLS requires client-side X.509 certificates to be deployed and installed on each AP6522 client bridge which are used for authentication. EAP-TLS is considered stronger than PEAP-MSCHAPv2 as a compromised password is not enough to break EAP-TLS. Passwords are only used to encrypt the X.509 client-side certificates for storage.

For ease of deployment the AP6522 client bridges support the X.509 client-side certificates using the PKCS #12 personal information exchange syntax standard. PKCS #12 defines a file format used to store private keys with accompanying public key certificates which are protected using a password based symmetric key. The PKCS #12 certificate files can be transferred to an AP6522 client bridge from a server using FTP, SFTP or TFTP protocols or from a Centralized Controller via MINT (future).

The following outlines the PKCS #12 client-side certificate requirements:

1. The exported PKCS #12 certificate must include the Private Key. When exporting the PKCS #12 certificate from Microsoft Certificate Services the option ***Allow private key to be exported*** must be selected.
2. The EAP username must be defined in the ***CN*** field in the ***Subject*** of the X.509 certificate using the ***username@example.com*** format.
3. The username in the CN field of the X.509 client-side certificate must match the username defined under radio 2 using the ***bridge eap username*** command. For example if the username in the CN field is set to ***client-bridge@tmelabs.local***, the eap username must also be set to ***client-bridge@tmelabs.local***.




```
!
! Radio 2 EAP-TLS Username Configuration Example
!
interface radio2
  rf-mode bridge
  bridge ssid MOTO-DOT1X
  bridge encryption-type ccmp
  bridge authentication-type eap
  bridge eap username client-bridge@tmelabs.local
  bridge eap password 0 hellomoto
  bridge eap type tls
```

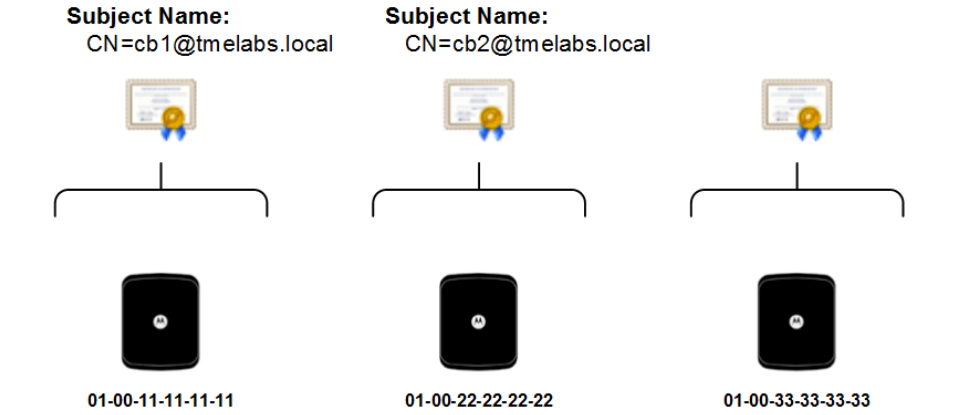
4. The password used to secure the PKCS #12 certificate during export must be defined under radio 2 using the ***bridge eap password*** command. The ***bridge eap password*** value must match the export password for the AP6522 client bridge to access the PKCS #12 file.

2.3.2.1 Common vs. Unique X.509 Certificates

AP6522 client bridges authenticating using EAP-TLS may be deployed with a common or unique X.509 client-side certificate. Both options are supported for standalone and centrally managed deployments and the X.509 client-side certificate deployment method selected will depend on how you wish to balance your security and administrative requirements.

Using unique X.509 client-side certificates greatly increases administrative / management overhead of the AP6522 client bridges but provides additional security. When unique X.509 certificates are deployed each AP6522 client bridge requires a unique PKCS #12 certificate file to be exported and uploaded onto each device. Additionally a unique ***bridge eap username*** and ***bridge eap password*** parameters may need to be defined as Device overrides if the individual X.509 certificates have unique CN fields or passwords.

The main advantage of deploying unique X.509 certificates is that it is very easy for an administrator to revoke a single X.509 certificate if an individual AP6522 client bridge is stolen or compromised. A second advantage is that each AP6522 client bridge can be assigned a unique username to simplify identification on the infrastructure Wireless LAN.



AP6522 Profile – Radio 2 Configuration

```
interface radio2
  rf-mode bridge
  bridge ssid MOTO-DOT1X
  bridge encryption-type ccmp
  bridge authentication-type eap
  bridge eap type tls
```

Device Overrides

```
!
ap6522 01-00-11-11-11-11
 use profile default-ap6522
 use rf-domain default
 hostname AP6522-CB1
 interface radio2
  bridge eap username cb1@tmelabs.local
  bridge eap password 0 hellomoto1
!
ap6522 01-00-22-22-22-22
 use profile default-ap6522
 use rf-domain default
 hostname AP6522-CB2
 interface radio2
  bridge eap username cb2@tmelabs.local
  bridge eap password 0 hellomoto2
!
ap6522 01-00-33-33-33-33
 use profile default-ap6522
 use rf-domain default
 hostname AP6522-CB3
 interface radio2
  bridge eap username cb3@tmelabs.local
  bridge eap password 0 hellomoto3
!
```

Figure 2.3.2.1-1 Unique X.509 Certificate Configuration Example

Using a common X.509 certificate greatly reduces the administrative / management overhead of the AP6522 client bridges as a single PKCS #12 certificate file can be exported and uploaded onto each device. As a single X.509 certificate is deployed the username in the CN field and export password will be the same on each AP6522 client bridge allowing the ***bridge eap username*** and ***bridge eap password*** parameters to be defined in the AP6522 Profile. No Device level overrides are required.

The main dis-advantage of deploying a common X.509 certificate is that an administrator cannot easily revoke the X.509 certificate if an individual AP6522 client bridge or X.509 client-side certificate is stolen or compromised. The X.509 certificates on all the AP6522 client bridges must be replaced with a new X.509 certificate before the compromised X.509 certificate can be revoked. Additionally as a single username is defined in the CN field, each AP6522 client bridge will use the same username which may make device identification more challenging on the infrastructure Wireless LAN.

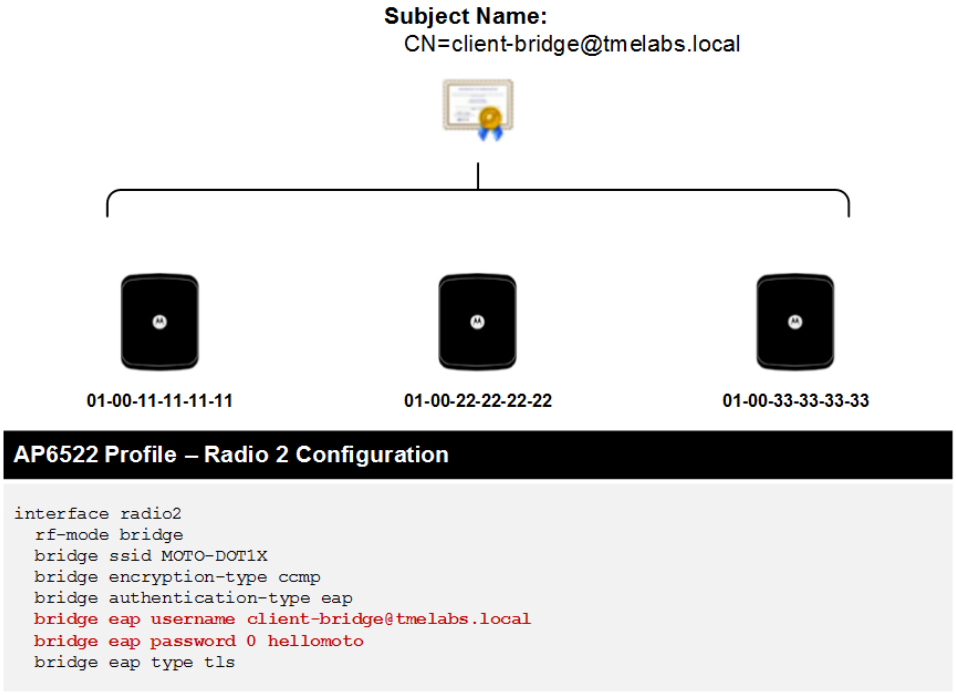


Figure 2.3.2.1-2 Common X.509 Certificate Configuration Example

☑	*It is strongly recommended that you enable Inline Password Encryption to protect the X.509 certificates export password. This provides an additional security level in the event that an AP6522 client bridge is stolen or compromised.*
☑	*For centrally managed deployments requiring unique X.509 client-certificates, the **bridge eap username** and **bridge eap password** parameters must be defined as Device overrides on the Centralized Controller prior to adoption. Radio 2 parameters cannot be currently learned from the new AP6522 client bridge during the initial adoption.*

2.3.2.2 Manual PKCS #12 Certificate Deployment

PKCS #12 certificate files can be manually transferred to the AP6522 client bridges using FTP, SFTP or TFTP protocols and must be installed prior to EAP-TLS being enabled on the client bridge radio. As a best practice it is recommended that the PKCS #12 file be transferred to the AP6522 client bridge as part of the staging process, however the PKCS #12 file can be transferred to AP6522 client bridges that are already associated and authenticated to the infrastructure Wireless LAN using PSK or PEAP-MSCHAPv2.

1A	**Transfer the PKCS #12 certificate file from a FTP server:**
Command Syntax:	
`copy ftp://<user>:<password>@<hostname|ip>[:port]/path/src-file dst-file`	
Example:	
`AP6522-CB1# copy ftp://ftpuser:hellomoto@192.168.10.10/cbcert.p12 wireless-bridge-certificate`	

1B	**Transfer the PKCS #12 certificate file from a SFTP server:**
Command Syntax:	
`copy sftp://<user>:<password>@<hostname|ip>[:port]/path/src-file dst-file`	
Example:	
`AP6522-CB1# copy sftp://ftpuser:hellomoto@192.168.10.10/cbcert.p12 wireless-bridge-certificate`	

1C	**Transfer the PKCS #12 certificate file from a TFTP server:**
Command Syntax:	
`copy tftp://<hostname|ip>[:port]/path/src-file dst-file`	
Example:	
`AP6522-CB1# copy tftp://192.168.10.10/cbcert.p12 wireless-bridge-certificate`	

*The PKCS #12 certificate file **MUST** be transferred to the AP6522 client bridge as the destination filename **wireless-bridge-certificate**.*

*Standard file extensions for PKCS #12 file are **.p12** or **.pfx**. PKCS #12 files created in OpenSSL use the **.p12** extension while PKCS #12 files created in Microsoft Certificate Services use the **.pfx** extension.*

2.4 Wired Host Authentication

Wired hosts connected to the Ge1 ports on the AP6522 client bridges can be optionally authenticated using IEEE 802.1X or MAC authentication.

IEEE 802.1X provides a standards port-based authentication mechanism to securely authenticate wired hosts connected to the Ge1 port against a backend RADIUS server using standard EAP methods such as EAP-TLS or Protected EAP (PEAP). When enabled the AP6522 client bridge will only permit access to the network after the wired host has successfully authenticated. Hosts that do not support 802.1X or fail authentication are denied access to the network. Depending on the operating system wired hosts can use either computer and/or user authentication.

MAC based authentication can be optionally enabled to authenticate wired hosts that do not support IEEE 802.1X. MAC authentication authenticates the wired hosts against the backend RADIUS server using the wired hosts MAC address as the username and password using PAP (default), CHAP, MSCHAP or MSCHAPv2 protocols. MAC authentication is enabled along with 802.1X and operates as a fall-thought authentication mechanism. The AP6522 client bridge will first attempt to authenticate the wired host using 802.1X and if unsuccessful will attempt MAC authentication.

802.1X and MAC port-based authentication is supported for both standalone and centrally managed AP6522 client bridge deployments. 802.1X supplicants are supporting by all current operating systems including commercial operating systems from Apple and Microsoft in addition to open source operating systems such as Linux. Commercial and open source third-party supplicants are also available for most current operating systems.

2.4.1 AAA Policy

An AAA Policy defines the RADIUS Authentication and RADIUS Accounting servers used by the AP6522 client bridges for 802.1X and MAC port-based authentication. Each AAA Policy includes one or more RADIUS Authentication server IP addresses / hostnames and shared secrets. The AAA Policy may also include one or more RADIUS Accounting Server IP addresses / hostnames and shared secrets if RADIUS Accounting is required along with the MAC address format and authentication protocol used for MAC authentication.

By default all RADIUS Authentication and Accounting requests will originate from the management IPv4 addresses assigned to each AP6522 client bridge. For centrally managed deployments you can optionally change the Proxy Mode for RADIUS Authentication and Accounting servers to ***Through Controller*** to originate RADIUS Authentication and Accounting requests from the Active Centralized Controller. This may be desirable for centrally managed deployments as it reduces the number of NAS Clients that need to be defined on the RADIUS servers.

2.4.1.1 802.1X Authentication

The following demonstrates how to create an AAA Policy required to enable 802.1X port-based authentication for standalone and centrally managed client bridge deployments:

1 Create a AAA Policy:

```
CBMGR-ACTIVE(config)# aaa-policy EXTERNAL-AAA
```

2 Define a RADIUS Server IP Address and Shared Secret:

```
CBMGR-ACTIVE(config-aaa-policy-EXTERNAL-AAA)# authentication server 1 host 192.168.10.6 secret hellomoto
```

3	Exit then Commit and Write the Changes:

```
CBMGR-ACTIVE(config-aaa-policy-EXTERNAL-AAA)# exit
CBMGR-ACTIVE(config)# commit write
```

By default RADIUS Authentication and Accounting packets will originate from the AP6522 Client Bridges. For centrally managed deployments you can optionally originate RADIUS Authentication and Accounting packets from the Active Centralized Controller by setting the RADIUS Proxy Mode for the Authentication Server to ***through-controller****.*

2.4.1.2 MAC Authentication

MAC authentication can use the same AAA Policy as 802.1X or a separate AAA Policy. If MAC based authentication is enabled you can optionally modify the formatting of the MAC address credentials as well as the authentication protocol. By default the AP6522 client bridge will attempt to MAC authenticate wired hosts using the PAP authentication protocol with the credentials using the pair-hyphen formatting in uppercase.

The MAC address format and authentication protocol used will be dependent on how the MAC address usernames and passwords are defined on the RADIUS server and the authentication protocol supported by the RADIUS server. Corporate security policies may also dictate that a strong authentication protocol such as MSCHAPv2 be utilized.

The following table provides an overview of the AAA Policy parameters which can be modified for MAC authentication:

Parameter	Description
authentication protocol [chap | mschap | mschapv2 | pap]	Determines the authentication protocol used for MAC authentication: ▪ ***chap*** – Use Challenge Handshake Authentication Protocol (CHAP) ▪ ***mschap*** – Use Use Microsoft Challenge Handshake Authentication Protocol (MSCHAP) ▪ ***mschapv2*** – Use Microsoft Challenge Handshake Authentication Protocol Version 2 (MS-CHAPv2) ▪ ***pap*** (default) – Use Password Authentication Protocol (PAP)
mac-address-format [middle-hyphen | no-delim | pair-colon | pair-hyphen | duad-dot] case [upper | lower]	Determines the formatting of the MAC address username and password forwarded to the RADIUS server: ▪ ***middle-hyphen*** – Formatted as AABBCC-DDEEFF ▪ ***no-delim*** – Formatted as AABBCCDDEEFF ▪ ***pair-colon*** – Formatted as AA:BB:CC:DD:EE:FF ▪ ***pair-hyphen*** (default) – Formatted as AA-BB-CC-DD-EE-FF ▪ ***quad-dot*** – Formatted as AABB.CCDD.EEFF

Table 2.4.1.2 – AAA Policy MAC Authentication Parameters

*By default MAC authentication will forward the MAC address using the **pair-hyphen** format using uppercase characters.*

2.4.2 AP6522 Profile

802.1X port based authentication is enabled on the AP6522 client bridges using the default or user defined AP6522 Profiles. 802.1X must be globally enabled and an AAA Policy assigned. 802.1X must also be enabled under the Ge1 port to define the 802.1X operating characteristics of the port and adjust timers.

The following table provides an overview of the 802.1X and MAC authentication parameters for the Ge1 port on the AP6522 client bridges:

Parameter	Description
dot1x authenticator host-mode [single-host | multi-host]	Determines how many wired hosts are supported by the EAPoL port: ▪ ***single-host*** (default) – A single wired host can be connected to the Ge1 port and authenticated. ▪ ***multi-host*** – Multiple wired hosts can be connected to the Ge1 port and are permitted once a single host has authenticated.
dot1x authenticator port-control [auto | forced-authorized | forced-unauthorized]	Determines the operating mode of the Ge1 port: ▪ ***auto*** – Enables EAPoL authentication on the Ge1 port preventing communications until the wired host has successfully authenticated. ▪ ***forced-authorized*** (default) – Places the Ge1 port into an authorized state permitting communications without EAPoL authentication. ▪ ***forced-unauthorized*** – Places the ge1 port into an unauthorized state preventing all communications.
dot1x authenticator reauthenticate	Determines if re-authentication is enabled on the Ge1 port. Disabled by default.
dot1x authenticator timeout reauth-period <1 - 65535>	Determines how many seconds before the wired host is re-authenticated. By default is set to 60 seconds.
dot1x authenticator timeout quiet-period <1 - 65535>	Determines how many seconds the Ge1 port waits after an unsuccessful authentication attempt before permitting the next authentication attempt. By default is set to 60 seconds.
mac-auth	Determines if MAC authentication is enabled on the Ge1 port. Disabled by default.

Table 2.4.2 – Profile / Device Ge1 802.1X Parameters

2.4.2.1 802.1X Authentication

The following demonstrates how to enable 802.1X port-based authentication on the Ge1 port for standalone and centrally managed client bridge deployments:

1 Access the AP6522 Profile:

```
CBMGR-ACTIVE(config)# profile ap6522 CB-AP6522
```

2 Map the AAA Policy and globally enable 802.1X Authentication:

```
CBMGR-ACTIVE(config-profile-CB-AP6522)# dot1x use aaa-policy EXTERNAL-AAA
CBMGR-ACTIVE(config-profile-CB-AP6522)# dot1x system-auth-control
```

3 Enable 802.1X Authentication on the Ge1 port:

```
CBMGR-ACTIVE(config-profile-CB-AP6522)# interface ge1
CBMGR-ACTIVE(config-profile-CB-AP6522-if-ge1)# dot1x authenticator port-control auto
```

4 Optionally enable 802.1X Re-authentication and define a Re-authentication Interval:

```
CBMGR-ACTIVE(config-profile-CB-AP6522-if-ge1)# dot1x authenticator reauthenticate
CBMGR-ACTIVE(config-profile-CB-AP6522-if-ge1)# dot1x authenticator timeout reauth-period 300
```

5 Exit then Commit and Write the Changes:

```
CBMGR-ACTIVE(config-profile-CB-AP6522-if-ge1)# exit
CBMGR-ACTIVE(config)# commit write
```

2.4.2.2 802.1X and MAC Authentication

The following demonstrates how to enable 802.1X and MAC port-based authentication on the Ge1 port for standalone and centrally managed client bridge deployments:

1 Access the AP6522 Profile:

```
CBMGR-ACTIVE(config)# profile ap6522 CB-AP6522
```

2 Map the AAA Policy and globally enable 802.1X Authentication:

```
CBMGR-ACTIVE(config-profile-CB-AP6522)# dot1x use aaa-policy EXTERNAL-AAA
CBMGR-ACTIVE(config-profile-CB-AP6522)# dot1x system-auth-control
```

3 Map the AAA Policy for MAC Authentication:

```
CBMGR-ACTIVE(config-profile-CB-AP6522)# mac-auth use aaa-policy EXTERNAL-AAA
```

4 Enable 802.1X and MAC Authentication on the Ge1 port:

```
CBMGR-ACTIVE(config-profile-CB-AP6522)# interface ge1
CBMGR-ACTIVE(config-profile-CB-AP6522-if-ge1)# dot1x authenticator port-control auto
CBMGR-ACTIVE(config-profile-CB-AP6522-if-ge1)# mac-auth
```

5 Optionally enable Re-Authentication and define a re-Authentication Interval:

```
CBMGR-ACTIVE(config-profile-CB-AP6522-if-ge1)# dot1x authenticator reauthenticate
CBMGR-ACTIVE(config-profile-CB-AP6522-if-ge1)# dot1x authenticator timeout reauth-period 300
```

6 Exit then Commit and Write the Changes:

```
CBMGR-ACTIVE(config-profile-CB-AP6522-if-ge1)# exit
CBMGR-ACTIVE(config)# commit write
```

3. Verification

3.1 Client Bridge

The following section provides an overview of the CLI commands which can be issued on the Centralized Controller or Standalone client bridge to verify AP6522 client bridge connectivity and operation.

3.1.1 Radio RF Mode

When a radio is configured for client bridge operation, the RF Mode will change from ***2.4GHz-wlan*** or ***5GHz-wlan*** to ***bridge***. You can verify the RF Mode is correct for the client bridge radio by issuing the ***show wireless radio*** command:

```
CBMGR-ACTIVE# show wireless radio on [<DEVICE> | <RF-DOMAIN>]
```

```
=============================================================================================
RADIO                RADIO-MAC             RF-MODE        STATE       CHANNEL    POWER #CLIENT
---------------------------------------------------------------------------------------------
AP6522-CB1:R1        B4-C7-99-E9-3C-A0 2.4GHz-wlan          Off   N/A (  smt)  0 (smt)       0
AP6522-CB1:R2        B4-C7-99-E9-26-C0      bridge           On    36 (  smt) 14 (smt)       0
=============================================================================================
Total number of radios displayed: 2
```

*If the state for radio 2 is **Off**, verify the country code has been defined in the RF Domain.*

3.1.2 Bridge Configuration

You can view the current client bridge configuration by issuing the ***show wireless bridge config*** command. The output of this command will display client bridges ***Hostname***, ***MAC Address***, ***Profile***, ***RF Domain***, ***SSID***, ***Band***, ***Encryption***, ***Authentication*** and ***EAP Username***:

```
CBMGR-ACTIVE# show wireless bridge config
```

```
-------------------------------------------------------------------------------------------------------------------------------------------------
 IDX       NAME             MAC               PROFILE         RF-DOMAIN     SSID        BAND      ENCRYPTION   AUTHENTICATION    EAP-USERNAME
-------------------------------------------------------------------------------------------------------------------------------------------------
  1   AP6522-CB1       B4-C7-99-63-D0-E4   CB-AP6522             STORE201    MOTO-DOT1X 2.4GHz/5GHz  ccmp         eap               EAPUSER
  2   AP6522-CB2       B4-C7-99-63-98-2C   CB-AP6522             STORE201    MOTO-DOT1X 2.4GHz/5GHz  ccmp         eap               EAPUSER
-------------------------------------------------------------------------------------------------------------------------------------------------
```

3.1.3 Bridge Candidate Access Points

When enabled the client bridge radio will scan its defined channel lists to locate the best candidate Access Point that is servicing the infrastructure Wireless LAN. You can view the candidate infrastructure Access Points as well as the infrastructure Access Point that the client bridge radio has selected by issuing the ***show wireless bridge candidate-ap*** command:

```
CBMGR-ACTIVE# show wireless bridge candidate-ap on <DEVICE>
```

```
Client Bridge Candidate APs:
  AP-MAC             BAND    CHANNEL SIGNAL(dbm) STATUS
  FC-0A-81-53-98-A2  2.4 GHz 11      -31         connected
  FC-0A-81-53-AE-D2  2.4 GHz 1       -42         unconnected
  FC-0A-81-53-B5-F2  2.4 GHz 6       -48         unconnected
  FC-0A-81-53-B2-62  2.4 GHz 11      -61         unconnected

Total number of radios displayed: 1
```

3.1.4 Bridge Hosts

You can view the client bridge host information by issuing the ***show wireless bridge*** hosts command. The output will display the client bridges ***Host MAC Address***, ***Bridge MAC Address***, ***IPv4 Address***, ***Bridging Status*** and ***Activity***:

```
CBMGR-ACTIVE# show wireless bridge hosts on [<DEVICE> | <RF-DOMAIN>]
```

```
================================================================================
HOST MAC             BRIDGE MAC         IP             BRIDGING STATUS ACTIVITY
                                                                    (sec ago)
--------------------------------------------------------------------------------
B4-C7-99-46-57-64    B4-C7-99-46-6B-C0 11.51.76.198    UP           00:00:00
00-90-FB-11-9B-42    B4-C7-99-46-6B-C1 11.51.76.136    UP           00:00:04
================================================================================
Total number of radios displayed: 2
```

*The **HOST MAC** field displays the real MAC addresses of the wired hosts while the **BRIDGE MAC** field displays the translated MAC addresses. The BRIDGE MAC field is based on the radio 2 base MAC address and will increment by 1 for each wired host connected to the client bridges Ge1 port.*

3.1.5 Bridge Statistics

You can view the client bridge RF statistics by issuing the ***show wireless bridge statistics rf*** command. The output will display the ***Signal***, ***Noise***, ***SNR***, ***TX/RX Rates***, ***Retries*** and ***Errors***:

```
CBMGR-ACTIVE# show wireless bridge statistics rf on [<DEVICE> | <RF-DOMAIN>]
```

```
=============================================================================================
LOCAL RADIO           CONNECTED AP     SIGNAL NOISE   SNR TX-RATE RX-RATE RETRY  ERRORS Q-
INDEX
                                       (dbm) (dbm)    db  (Mbps)  (Mbps)   AVG   (pps)     (%)
---------------------------------------------------------------------------------------------
AP6522-CB2:R2        FC-0A-81-53-98-A2   -28  -102    74      53      26     0       4     100
AP6522-CB2:R2        FC-0A-81-53-AE-D2   -43  -102    59      53      27     0       4     100
=============================================================================================
Total number of radios displayed: 2
```

You can view the client bridge radio traffic statistics by issuing the ***show wireless bridge statistics*** traffic command. The output will display ***TX/RX Bytes***, ***TX/RX Packets***, ***TX/RX Bits/Second*** and ***Dropped Packets***:

```
HOSTNAME# show wireless bridge statistics traffic on [<DEVICE> | <RF-DOMAIN>]
```

```
=============================================================================================================================
LOCAL RADIO           CONNECTED AP             Tx          Rx          Tx          Rx      Tx      Rx  T-INDEX      Dropped
                                             bytes       bytes       pkts        pkts     bps     bps     (%)         pkts
-----------------------------------------------------------------------------------------------------------------------------
AP6522-CB2:R2        FC-0A-81-53-98-A2      187451      491222        1123        2489     0 k     0 k       0            0
AP6522-CB2:R2        FC-0A-81-53-AE-D2      219924      644341        1305        3081     0 k     0 k       0            0
=============================================================================================================================
Total number of radios displayed: 2
```

3.1.6 Switched Virtual Interfaces

Once a client bridge has connected to a portal it will be manageable via the switched virtual interface (SVI) assigned to the bridged VLAN ID. The IPv4 address statically or dynamically assigned to the client bridge can be verified by issuing the ***show ip interface brief*** command:

```
CBMGR-ACTIVE# show ip interface brief on <DEVICE>
```

```
------------------------------------------------------------------------------
 INTERFACE          IP-ADDRESS/MASK            TYPE        STATUS   PROTOCOL
------------------------------------------------------------------------------
 vlan1              192.168.12.111/24(DHCP)    primary     UP       up
------------------------------------------------------------------------------
```

3.2 Wired Host Authentication

The following section provides an overview of the CLI commands which can be issued on the Centralized Controller or Standalone client bridge to verify 802.1X and MAC port-based authentication.

3.2.1 802.1X Authentication

You can view the 802.1X configuration and authentication state by issuing the ***show dot1x*** command. The output of the command will also display the current authentication state of the wired host in addition to the authorization state of the Ge1 port. The ***Port Status*** will display ***AUTHORIZED*** if the wired host has successfully authenticated and ***Not Authorized*** if the wired host is not yet authenticated or has failed 802.1X authentication:

```
NX9500-ACTIVE# show dot1x on <DEVICE>
```

```
SysAuthControl is enabled
Guest-Vlan is disabled
AAA-Policy is EXTERNAL-AAA

Dot1x info for interface GE1
-----------------------------------
Supplicant MAC B4-99-BA-F0-87-B7
  Auth SM State = AUTHENTICATED
  Bend SM State = IDLE
  Port Status   = AUTHORIZED
  Host Mode     = SINGLE
  Auth Vlan     = None
  Guest Vlan    = None
```

3.2.2 MAC Authentication

You can view the MAC authentication configuration and authentication state by issuing the ***show mac-auth*** command. The output of the command will also display the current authentication state of the wired host, the authorization state of the Ge1 port and the wired hosts MAC address. The ***Port Status*** will display ***AUTHORIZED*** if the wired host has successfully authenticated and ***Not Authorized*** if the wired host is not yet authenticated or has failed MAC authentication:

```
NX9500-ACTIVE# show mac-auth on <DEVICE>
```

```
AAA-Policy is EXTERNAL-AAA

Mac Auth info for interface GE1
-----------------------------------
 Mac Auth Enabled
 Mac Auth Authorized
Client MAC B4-99-BA-F0-87-B7
```

3.3 Wireless LAN Infrastructure

When connected to an infrastructure Access Point the client bridge radio in addition to each wired device connected to the client bridges Ge1 port will be displayed as Wireless LAN clients on the Wireless LAN infrastructure.

When viewing the Wireless LAN clients on the Wireless LAN infrastructure it's important to note that the actual MAC address of each wired device will not be displayed as the wired devices real MAC addresses are masqueraded. The AP6522 client bridge will translate the real MAC address of each wired device to a pseudo MAC address that is based on the base MAC address of the client bridge radio. The base MAC address will be used for the client bridge connection to the Wireless LAN while each wired MAC address will increment the based MAC address by 1 (16 total).

```
NX9500-ACTIVE# show wireless client on [<DEVICE> | <RF-DOMAIN>]
```

```
================================================================================================================================================
MAC                      IPv4    AP-NAME          WLAN                VLAN         STATE AUTH  HOSTNAME          USER NAME
------------------------------------------------------------------------------------------------------------------------------------------------
B4-C7-99-E9-25-E0   192.168.12.216 AP8222-4         MOTO-DOT1X          12      Data-Ready eap   AP6522-CB1        eapuser1
B4-C7-99-E9-25-E1   192.168.12.215 AP8222-4         MOTO-DOT1X          12      Data-Ready eap   KVBF73-LABPC      eapuser1
================================================================================================================================================
Total number of wireless clients displayed: 2
```

*The real MAC address of the wired hosts can be determined by issuing the **show bridge hosts** command on the Centralized Controller or directly on a standalone AP6522 client bridge.*

4. Appendix

4.1 Staging Config File Examples

4.1.1 WPA2 Pre-Share Key Example

```
!### show running-config
!
! Configuration of AP6522 version 5.5.20.0-003D
!
!
version 2.3
!
!
ip access-list BROADCAST-MULTICAST-CONTROL
 permit tcp any any rule-precedence 10 rule-description "permit all TCP traffic"
 permit udp any eq 67 any eq dhcpc rule-precedence 11 rule-description "permit DHCP replies"
 deny udp any range 137 138 any range 137 138 rule-precedence 20 rule-description "deny windows netbios"
 deny ip any 224.0.0.0/4 rule-precedence 21 rule-description "deny IP multicast"
 deny ip any host 255.255.255.255 rule-precedence 22 rule-description "deny IP local broadcast"
 permit ip any any rule-precedence 100 rule-description "permit all IP traffic"
!
mac access-list PERMIT-ARP-AND-IPv4
 permit any any type ip rule-precedence 10 rule-description "permit all IPv4 traffic"
 permit any any type arp rule-precedence 20 rule-description "permit all ARP traffic"
!
ip snmp-access-list default
 permit any
!
firewall-policy default
 no ip dos smurf
 no ip dos twinge
 no ip dos invalid-protocol
 no ip dos router-advt
 no ip dos router-solicit
 no ip dos option-route
 no ip dos ascend
 no ip dos chargen
 no ip dos fraggle
 no ip dos snork
 no ip dos ftp-bounce
 no ip dos tcp-intercept
 no ip dos broadcast-multicast-icmp
 no ip dos land
 no ip dos tcp-xmas-scan
 no ip dos tcp-null-scan
 no ip dos winnuke
 no ip dos tcp-fin-scan
 no ip dos udp-short-hdr
 no ip dos tcp-post-syn
 no ip dos tcphdrfrag
```

```
 no ip dos ip-ttl-zero
 no ip dos ipspoof
 no ip dos tcp-bad-sequence
 no ip dos tcp-sequence-past-window
 no ip-mac conflict
 no ip-mac routing conflict
 no firewall enable
 no stateful-packet-inspection-l2
!
!
mint-policy global-default
!
meshpoint-qos-policy default
!
wlan-qos-policy default
 qos trust dscp
 qos trust wmm
!
radio-qos-policy default
!
!
management-policy default
 https server
 ssh
 user admin password 0 hellomoto role superuser access all
 no snmp-server manager v3
 snmp-server community 0 public ro
 snmp-server user snmptrap v3 encrypted des auth md5 0 Zebra
 snmp-server user snmpmanager v3 encrypted des auth md5 0 Zebra
!
l2tpv3 policy default
!
profile ap6522 default-ap6522
 mint level 1 area-id 65535
 ip name-server 192.168.10.6
 ip domain-name tmelabs.local
 autoinstall configuration
 autoinstall firmware
 crypto ikev1 policy ikev1-default
  isakmp-proposal default encryption aes-256 group 2 hash sha
 crypto ikev2 policy ikev2-default
  isakmp-proposal default encryption aes-256 group 2 hash sha
 crypto ipsec transform-set default esp-aes-256 esp-sha-hmac
 crypto ikev1 remote-vpn
 crypto ikev2 remote-vpn
 crypto auto-ipsec-secure
 crypto load-management
 crypto remote-vpn-client
 interface radio1
  shutdown
 interface radio2
  rf-mode bridge
  bridge ssid TMELABS-PSK
  bridge encryption-type ccmp
```

```
  bridge wpa-wpa2 psk hellomoto
 interface ge1
  ip dhcp trust
  qos trust dscp
  qos trust 802.1p
 interface vlan1
  ip address dhcp
  ip dhcp client request options all
 interface pppoe1
 use firewall-policy default
 ntp server 192.168.10.1
 service pm sys-restart
 router ospf
!
rf-domain default
 country-code us
!
self
 use profile default-ap6522
 use rf-domain default
 hostname AP6522-CB1
 logging on
 logging console warnings
 logging buffered warnings
!
!
end
```

4.1.2 WPA2 PEAP MSCHAPv2 Example

```
!### show running-config
!
! Configuration of AP6522 version 5.5.20.0-003D
!
!
version 2.3
!
!
ip access-list BROADCAST-MULTICAST-CONTROL
 permit tcp any any rule-precedence 10 rule-description "permit all TCP traffic"
 permit udp any eq 67 any eq dhcpc rule-precedence 11 rule-description "permit DHCP replies"
 deny udp any range 137 138 any range 137 138 rule-precedence 20 rule-description "deny windows netbios"
 deny ip any 224.0.0.0/4 rule-precedence 21 rule-description "deny IP multicast"
 deny ip any host 255.255.255.255 rule-precedence 22 rule-description "deny IP local broadcast"
 permit ip any any rule-precedence 100 rule-description "permit all IP traffic"
!
mac access-list PERMIT-ARP-AND-IPv4
 permit any any type ip rule-precedence 10 rule-description "permit all IPv4 traffic"
 permit any any type arp rule-precedence 20 rule-description "permit all ARP traffic"
!
ip snmp-access-list default
 permit any
!
```

```
firewall-policy default
 no ip dos smurf
 no ip dos twinge
 no ip dos invalid-protocol
 no ip dos router-advt
 no ip dos router-solicit
 no ip dos option-route
 no ip dos ascend
 no ip dos chargen
 no ip dos fraggle
 no ip dos snork
 no ip dos ftp-bounce
 no ip dos tcp-intercept
 no ip dos broadcast-multicast-icmp
 no ip dos land
 no ip dos tcp-xmas-scan
 no ip dos tcp-null-scan
 no ip dos winnuke
 no ip dos tcp-fin-scan
 no ip dos udp-short-hdr
 no ip dos tcp-post-syn
 no ip dos tcphdrfrag
 no ip dos ip-ttl-zero
 no ip dos ipspoof
 no ip dos tcp-bad-sequence
 no ip dos tcp-sequence-past-window
 no ip-mac conflict
 no ip-mac routing conflict
 no firewall enable
 no stateful-packet-inspection-l2
!
!
mint-policy global-default
!
meshpoint-qos-policy default
!
wlan-qos-policy default
 qos trust dscp
 qos trust wmm
!
radio-qos-policy default
!
!
management-policy default
 https server
 ssh
 user admin password 0 hellomoto role superuser access all
 no snmp-server manager v3
 snmp-server community 0 public ro
 snmp-server user snmptrap v3 encrypted des auth md5 0 Zebra
 snmp-server user snmpmanager v3 encrypted des auth md5 0 Zebra
!
l2tpv3 policy default
!
```

```
profile ap6522 default-ap6522
 mint level 1 area-id 65535
 ip name-server 192.168.10.6
 ip domain-name tmelabs.local
 autoinstall configuration
 autoinstall firmware
 crypto ikev1 policy ikev1-default
  isakmp-proposal default encryption aes-256 group 2 hash sha
 crypto ikev2 policy ikev2-default
  isakmp-proposal default encryption aes-256 group 2 hash sha
 crypto ipsec transform-set default esp-aes-256 esp-sha-hmac
 crypto ikev1 remote-vpn
 crypto ikev2 remote-vpn
 crypto auto-ipsec-secure
 crypto load-management
 crypto remote-vpn-client
 interface radio1
  shutdown
 interface radio2
  rf-mode bridge
  bridge ssid TMELABS-DOT1X
  bridge encryption-type ccmp
  bridge authentication-type eap
  bridge eap username eapuser1
  bridge eap password 0 hellomoto
 interface ge1
  ip dhcp trust
  qos trust dscp
  qos trust 802.1p
 interface vlan1
  ip address dhcp
  ip dhcp client request options all
 interface pppoe1
 use firewall-policy default
 ntp server 192.168.10.1
 service pm sys-restart
 router ospf
!
rf-domain default
 country-code us
!
self
 use profile default-ap6522
 use rf-domain default
 hostname AP6522-CB1
 logging on
 logging console warnings
 logging buffered warnings
!
!
end
```

4.1.3 WPA2 EAP-TLS Example

```
!### show running-config
!
! Configuration of AP6522 version 5.5.20.0-004D
!
!
version 2.3
!
!
ip access-list BROADCAST-MULTICAST-CONTROL
 permit tcp any any rule-precedence 10 rule-description "permit all TCP traffic"
 permit udp any eq 67 any eq dhcpc rule-precedence 11 rule-description "permit DHCP replies"
 deny udp any range 137 138 any range 137 138 rule-precedence 20 rule-description "deny windows netbios"
 deny ip any 224.0.0.0/4 rule-precedence 21 rule-description "deny IP multicast"
 deny ip any host 255.255.255.255 rule-precedence 22 rule-description "deny IP local broadcast"
 permit ip any any rule-precedence 100 rule-description "permit all IP traffic"
!
mac access-list PERMIT-ARP-AND-IPv4
 permit any any type ip rule-precedence 10 rule-description "permit all IPv4 traffic"
 permit any any type arp rule-precedence 20 rule-description "permit all ARP traffic"
!
ip snmp-access-list default
 permit any
!
firewall-policy default
 no ip dos smurf
 no ip dos twinge
 no ip dos invalid-protocol
 no ip dos router-advt
 no ip dos router-solicit
 no ip dos option-route
 no ip dos ascend
 no ip dos chargen
 no ip dos fraggle
 no ip dos snork
 no ip dos ftp-bounce
 no ip dos tcp-intercept
 no ip dos broadcast-multicast-icmp
 no ip dos land
 no ip dos tcp-xmas-scan
 no ip dos tcp-null-scan
 no ip dos winnuke
 no ip dos tcp-fin-scan
 no ip dos udp-short-hdr
 no ip dos tcp-post-syn
 no ip dos tcphdrfrag
 no ip dos ip-ttl-zero
 no ip dos ipspoof
 no ip dos tcp-bad-sequence
 no ip dos tcp-sequence-past-window
 no ip-mac conflict
 no ip-mac routing conflict
 no firewall enable
 no stateful-packet-inspection-l2
```

```
!
!
mint-policy global-default
!
meshpoint-qos-policy default
!
wlan-qos-policy default
 qos trust dscp
 qos trust wmm
!
radio-qos-policy default
!
!
management-policy default
 https server
 ssh
 user admin password 0 hellomoto role superuser access all
 no snmp-server manager v3
 snmp-server community 0 public ro
 snmp-server user snmptrap v3 encrypted des auth md5 0 Zebra
 snmp-server user snmpmanager v3 encrypted des auth md5 0 Zebra
!
l2tpv3 policy default
!
profile ap6522 default-ap6522
 mint level 1 area-id 65535
 ip name-server 192.168.10.6
 ip domain-name tmelabs.local
 autoinstall configuration
 autoinstall firmware
 crypto ikev1 policy ikev1-default
  isakmp-proposal default encryption aes-256 group 2 hash sha
 crypto ikev2 policy ikev2-default
  isakmp-proposal default encryption aes-256 group 2 hash sha
 crypto ipsec transform-set default esp-aes-256 esp-sha-hmac
 crypto ikev1 remote-vpn
 crypto ikev2 remote-vpn
 crypto auto-ipsec-secure
 crypto load-management
 crypto remote-vpn-client
 interface radio1
  shutdown
 interface radio2
  rf-mode bridge
  bridge ssid TMELABS-DOT1X
  bridge encryption-type ccmp
  bridge authentication-type eap
  bridge eap username eapuser1@tmelabs.local
  bridge eap password 0 hellomoto
  bridge eap type tls
 interface ge1
  ip dhcp trust
  qos trust dscp
  qos trust 802.1p
```

```
 interface vlan1
  ip address dhcp
  ip dhcp client request options all
 interface pppoe1
 use firewall-policy default
 ntp server 192.168.10.1
 service pm sys-restart
 router ospf
!
rf-domain default
 country-code us
!
self
 use profile default-ap6522
 use rf-domain default
 hostname AP6522-CB1
 logging on
 logging console warnings
 logging buffered warnings
!
!
end
```

4.2 Centrally Managed Configuration File Examples

4.2.1 WPA2 Pre-Share Key Example

```
!### show running-config
!
! Configuration of NX9000 version 5.5.20.0-004D
!
!
version 2.3
!
!
ip access-list BROADCAST-MULTICAST-CONTROL
 permit tcp any any rule-precedence 10 rule-description "permit all TCP traffic"
 permit udp any eq 67 any eq dhcpc rule-precedence 11 rule-description "permit DHCP replies"
 deny udp any range 137 138 any range 137 138 rule-precedence 20 rule-description "deny windows netbios"
 deny ip any 224.0.0.0/4 rule-precedence 21 rule-description "deny IP multicast"
 deny ip any host 255.255.255.255 rule-precedence 22 rule-description "deny IP local broadcast"
 permit ip any any rule-precedence 100 rule-description "permit all IP traffic"
!
mac access-list PERMIT-ARP-AND-IPv4
 permit any any type ip rule-precedence 10 rule-description "permit all IPv4 traffic"
 permit any any type arp rule-precedence 20 rule-description "permit all ARP traffic"
!
ip snmp-access-list default
 permit any
!
firewall-policy default
 no ip dos smurf
 no ip dos twinge
 no ip dos invalid-protocol
 no ip dos router-advt
 no ip dos router-solicit
 no ip dos option-route
 no ip dos ascend
 no ip dos chargen
 no ip dos fraggle
 no ip dos snork
 no ip dos ftp-bounce
 no ip dos tcp-intercept
 no ip dos broadcast-multicast-icmp
 no ip dos land
 no ip dos tcp-xmas-scan
 no ip dos tcp-null-scan
 no ip dos winnuke
 no ip dos tcp-fin-scan
 no ip dos udp-short-hdr
 no ip dos tcp-post-syn
 no ip dos tcphdrfrag
 no ip dos ip-ttl-zero
 no ip dos ipspoof
 no ip dos tcp-bad-sequence
 no ip dos tcp-sequence-past-window
 no ip-mac conflict
 no ip-mac routing conflict
```

```
 no firewall enable
 no stateful-packet-inspection-l2
!
!
mint-policy global-default
!
meshpoint-qos-policy default
!
wlan-qos-policy default
 qos trust dscp
 qos trust wmm
!
radio-qos-policy default
!
auto-provisioning-policy DATACENTER
 adopt ap6532 precedence 1 profile STORES-AP6532 rf-domain STORE1 ip 192.168.21.0/24
!
management-policy CENTRALIZED-CONTROLLERS
 https server
 ssh
 user admin password 0 hellomoto role superuser access all
 no snmp-server manager v3
 snmp-server community 0 public ro
 snmp-server user snmptrap v3 encrypted des auth md5 0 Zebra
 snmp-server user snmpmanager v3 encrypted des auth md5 0 Zebra
!
management-policy CLIENT-BRIDGES
 no http server
 ssh
 user admin password 0 hellomoto role superuser access all
 no snmp-server manager v3
!
l2tpv3 policy default
!
profile nx9000 DATACENTER-NX9500
 ip name-server 192.168.10.6
 ip domain-name tmelabs.local
 no autoinstall configuration
 no autoinstall firmware
 crypto ikev1 policy ikev1-default
  isakmp-proposal default encryption aes-256 group 2 hash sha
 crypto ikev2 policy ikev2-default
  isakmp-proposal default encryption aes-256 group 2 hash sha
 crypto ipsec transform-set default esp-aes-256 esp-sha-hmac
 crypto ikev1 remote-vpn
 crypto ikev2 remote-vpn
 crypto auto-ipsec-secure
 crypto load-management
 crypto remote-vpn-client
 interface ge1
  description UPLINK
  ip dhcp trust
  qos trust dscp
  qos trust 802.1p
```

```
 interface xge1
  ip dhcp trust
  qos trust dscp
  qos trust 802.1p
 interface ge2
  ip dhcp trust
  qos trust dscp
  qos trust 802.1p
 interface xge2
  ip dhcp trust
  qos trust dscp
  qos trust 802.1p
 interface xge3
  ip dhcp trust
  qos trust dscp
  qos trust 802.1p
 interface xge4
  ip dhcp trust
  qos trust dscp
  qos trust 802.1p
 use management-policy CENTRALIZED-CONTROLLERS
 use firewall-policy default
 use auto-provisioning-policy DATACENTER
 ntp server 192.168.10.1
service pm sys-restart
!
profile ap6522 CB-AP6522
 mint level 1 area-id 65535
 ip name-server 192.168.10.6
 ip domain-name tmelabs.local
 autoinstall configuration
 autoinstall firmware
 crypto ikev1 policy ikev1-default
  isakmp-proposal default encryption aes-256 group 2 hash sha
 crypto ikev2 policy ikev2-default
  isakmp-proposal default encryption aes-256 group 2 hash sha
 crypto ipsec transform-set default esp-aes-256 esp-sha-hmac
 crypto ikev1 remote-vpn
 crypto ikev2 remote-vpn
 crypto auto-ipsec-secure
 crypto load-management
 crypto remote-vpn-client
 interface radio1
  shutdown
 interface radio2
  rf-mode bridge
  bridge ssid TMELABS-PSK
  bridge encryption-type ccmp
  bridge wpa-wpa2 psk hellomoto
 interface ge1
  ip dhcp trust
  qos trust dscp
  qos trust 802.1p
 interface vlan1
```

```
  ip address dhcp
  ip dhcp client request options all
 interface pppoe1
 use management-policy CLIENT-BRIDGES
 use firewall-policy default
 ntp server 192.168.10.1
 controller host 192.168.20.90 pool 1 level 2
 controller host 192.168.20.91 pool 2 level 2
 service pm sys-restart
 router ospf
!
rf-domain DATACENTER
location "Johnson City TN"
 contact kmarshall@Zebrasolutions.com
 timezone EST5EDT
 no country-code
!
rf-domain STORE201
 location "Johnson City TN"
 contact kmarshall@Zebrasolutions.com
 timezone EST5EDT
 country-code us
 controller-managed
!
!
nx9000 00-0C-29-91-84-D1
 use profile DATACENTER-NX9500
 use rf-domain DATACENTER
 hostname CBMGR-ACTIVE
 license AAP e3b53caa9ea6ec4fbb80bf6237a540ec514ddfbde1d0fa435d03bcdd648a2c6a8b682f0679a9b2fc
 ip default-gateway 192.168.20.1
 interface vlan1
  description MANAGEMENT
  ip address 192.168.20.90/24
 cluster name DATACENTER1
 cluster mode active
 cluster member ip 192.168.20.91 level 2
 cluster master-priority 254
 logging on
 logging console warnings
 logging buffered warnings
!
nx9000 00-0C-29-98-CB-CD
 use profile DATACENTER-NX9500
 use rf-domain DATACENTER
 hostname CBMGR-STANDBY
 ip default-gateway 192.168.20.1
 interface vlan1
  description MANAGEMENT
  ip address 192.168.20.91/24
 cluster name DATACENTER1
 cluster mode standby
 cluster member ip 192.168.20.90 level 2
 cluster master-priority 128
```

```
 logging on
 logging console warnings
 logging buffered warnings
!
!
End
```

4.2.2 WPA2 PEAP MSCHAPv2 Example

```
!### show running-config
!
! Configuration of NX9000 version 5.5.20.0-004D
!
!
version 2.3
!
!
ip access-list BROADCAST-MULTICAST-CONTROL
 permit tcp any any rule-precedence 10 rule-description "permit all TCP traffic"
 permit udp any eq 67 any eq dhcpc rule-precedence 11 rule-description "permit DHCP replies"
 deny udp any range 137 138 any range 137 138 rule-precedence 20 rule-description "deny windows netbios"
 deny ip any 224.0.0.0/4 rule-precedence 21 rule-description "deny IP multicast"
 deny ip any host 255.255.255.255 rule-precedence 22 rule-description "deny IP local broadcast"
 permit ip any any rule-precedence 100 rule-description "permit all IP traffic"
!
mac access-list PERMIT-ARP-AND-IPv4
 permit any any type ip rule-precedence 10 rule-description "permit all IPv4 traffic"
 permit any any type arp rule-precedence 20 rule-description "permit all ARP traffic"
!
ip snmp-access-list default
 permit any
!
firewall-policy default
 no ip dos smurf
 no ip dos twinge
 no ip dos invalid-protocol
 no ip dos router-advt
 no ip dos router-solicit
 no ip dos option-route
 no ip dos ascend
 no ip dos chargen
 no ip dos fraggle
 no ip dos snork
 no ip dos ftp-bounce
 no ip dos tcp-intercept
 no ip dos broadcast-multicast-icmp
 no ip dos land
 no ip dos tcp-xmas-scan
 no ip dos tcp-null-scan
 no ip dos winnuke
 no ip dos tcp-fin-scan
 no ip dos udp-short-hdr
 no ip dos tcp-post-syn
```

```
 no ip dos tcphdrfrag
 no ip dos ip-ttl-zero
 no ip dos ipspoof
 no ip dos tcp-bad-sequence
 no ip dos tcp-sequence-past-window
 no ip-mac conflict
 no ip-mac routing conflict
 no firewall enable
 no stateful-packet-inspection-l2
!
!
mint-policy global-default
!
meshpoint-qos-policy default
!
wlan-qos-policy default
 qos trust dscp
 qos trust wmm
!
radio-qos-policy default
!
auto-provisioning-policy DATACENTER
 adopt ap6532 precedence 1 profile STORES-AP6532 rf-domain STORE1 ip 192.168.21.0/24
!
management-policy CENTRALIZED-CONTROLLERS
 https server
 ssh
 user admin password 0 hellomoto role superuser access all
 no snmp-server manager v3
 snmp-server community 0 public ro
 snmp-server user snmptrap v3 encrypted des auth md5 0 Zebra
 snmp-server user snmpmanager v3 encrypted des auth md5 0 Zebra
!
management-policy CLIENT-BRIDGES
 no http server
 ssh
 user admin password 0 hellomoto role superuser access all
 no snmp-server manager v3
!
l2tpv3 policy default
!
profile nx9000 DATACENTER-NX9500
 ip name-server 192.168.10.6
 ip domain-name tmelabs.local
 no autoinstall configuration
 no autoinstall firmware
 crypto ikev1 policy ikev1-default
  isakmp-proposal default encryption aes-256 group 2 hash sha
 crypto ikev2 policy ikev2-default
  isakmp-proposal default encryption aes-256 group 2 hash sha
 crypto ipsec transform-set default esp-aes-256 esp-sha-hmac
 crypto ikev1 remote-vpn
 crypto ikev2 remote-vpn
 crypto auto-ipsec-secure
```

```
 crypto load-management
 crypto remote-vpn-client
 interface ge1
  description UPLINK
  ip dhcp trust
  qos trust dscp
  qos trust 802.1p
 interface xge1
  ip dhcp trust
  qos trust dscp
  qos trust 802.1p
 interface ge2
  ip dhcp trust
  qos trust dscp
  qos trust 802.1p
 interface xge2
  ip dhcp trust
  qos trust dscp
  qos trust 802.1p
 interface xge3
  ip dhcp trust
  qos trust dscp
  qos trust 802.1p
 interface xge4
  ip dhcp trust
  qos trust dscp
  qos trust 802.1p
 use management-policy CENTRALIZED-CONTROLLERS
 use firewall-policy default
 use auto-provisioning-policy DATACENTER
 ntp server 192.168.10.1
service pm sys-restart
!
profile ap6522 CB-AP6522
 mint level 1 area-id 65535
 ip name-server 192.168.10.6
 ip domain-name tmelabs.local
 autoinstall configuration
 autoinstall firmware
 crypto ikev1 policy ikev1-default
  isakmp-proposal default encryption aes-256 group 2 hash sha
 crypto ikev2 policy ikev2-default
  isakmp-proposal default encryption aes-256 group 2 hash sha
 crypto ipsec transform-set default esp-aes-256 esp-sha-hmac
 crypto ikev1 remote-vpn
 crypto ikev2 remote-vpn
 crypto auto-ipsec-secure
 crypto load-management
 crypto remote-vpn-client
 interface radio1
  shutdown
 interface radio2
  rf-mode bridge
  bridge ssid TMELABS-DOT1X
```

```
  bridge encryption-type ccmp
  bridge authentication-type eap
  bridge eap username eapuser1
  bridge eap password 0 hellomoto
 interface ge1
  ip dhcp trust
  qos trust dscp
  qos trust 802.1p
 interface vlan1
  ip address dhcp
  ip dhcp client request options all
 interface pppoe1
 use management-policy CLIENT-BRIDGES
 use firewall-policy default
 ntp server 192.168.10.1
 controller host 192.168.20.90 pool 1 level 2
 controller host 192.168.20.91 pool 2 level 2
 service pm sys-restart
 router ospf
!
rf-domain DATACENTER
location "Johnson City TN"
 contact kmarshall@Zebrasolutions.com
 timezone EST5EDT
 no country-code
!
rf-domain STORE201
 location "Johnson City TN"
 contact kmarshall@Zebrasolutions.com
 timezone EST5EDT
 country-code us
 controller-managed
!
!
nx9000 00-0C-29-91-84-D1
 use profile DATACENTER-NX9500
 use rf-domain DATACENTER
 hostname CBMGR-ACTIVE
 license AAP e3b53caa9ea6ec4fbb80bf6237a540ec514ddfbde1d0fa435d03bcdd648a2c6a8b682f0679a9b2fc
 ip default-gateway 192.168.20.1
 interface vlan1
  description MANAGEMENT
  ip address 192.168.20.90/24
 cluster name DATACENTER1
 cluster mode active
 cluster member ip 192.168.20.91 level 2
 cluster master-priority 254
 logging on
 logging console warnings
 logging buffered warnings
!
nx9000 00-0C-29-98-CB-CD
 use profile DATACENTER-NX9500
 use rf-domain DATACENTER
```

```
 hostname CBMGR-STANDBY
 ip default-gateway 192.168.20.1
 interface vlan1
  description MANAGEMENT
  ip address 192.168.20.91/24
 cluster name DATACENTER1
 cluster mode standby
 cluster member ip 192.168.20.90 level 2
 cluster master-priority 128
 logging on
 logging console warnings
 logging buffered warnings
!
!
end
```

4.2.3 WPA2 EAP-TLS Example

```
!### show running-config
!
! Configuration of NX9000 version 5.5.20.0-004D
!
!
version 2.3
!
!
ip access-list BROADCAST-MULTICAST-CONTROL
 permit tcp any any rule-precedence 10 rule-description "permit all TCP traffic"
 permit udp any eq 67 any eq dhcpc rule-precedence 11 rule-description "permit DHCP replies"
 deny udp any range 137 138 any range 137 138 rule-precedence 20 rule-description "deny windows netbios"
 deny ip any 224.0.0.0/4 rule-precedence 21 rule-description "deny IP multicast"
 deny ip any host 255.255.255.255 rule-precedence 22 rule-description "deny IP local broadcast"
 permit ip any any rule-precedence 100 rule-description "permit all IP traffic"
!
mac access-list PERMIT-ARP-AND-IPv4
 permit any any type ip rule-precedence 10 rule-description "permit all IPv4 traffic"
 permit any any type arp rule-precedence 20 rule-description "permit all ARP traffic"
!
ip snmp-access-list default
 permit any
!
firewall-policy default
 no ip dos smurf
 no ip dos twinge
 no ip dos invalid-protocol
 no ip dos router-advt
 no ip dos router-solicit
 no ip dos option-route
 no ip dos ascend
 no ip dos chargen
 no ip dos fraggle
 no ip dos snork
 no ip dos ftp-bounce
```

```
 no ip dos tcp-intercept
 no ip dos broadcast-multicast-icmp
 no ip dos land
 no ip dos tcp-xmas-scan
 no ip dos tcp-null-scan
 no ip dos winnuke
 no ip dos tcp-fin-scan
 no ip dos udp-short-hdr
 no ip dos tcp-post-syn
 no ip dos tcphdrfrag
 no ip dos ip-ttl-zero
 no ip dos ipspoof
 no ip dos tcp-bad-sequence
 no ip dos tcp-sequence-past-window
 no ip-mac conflict
 no ip-mac routing conflict
 no firewall enable
 no stateful-packet-inspection-l2
!
!
mint-policy global-default
!
meshpoint-qos-policy default
!
wlan-qos-policy default
 qos trust dscp
 qos trust wmm
!
radio-qos-policy default
!
auto-provisioning-policy DATACENTER
 adopt ap6532 precedence 1 profile STORES-AP6532 rf-domain STORE1 ip 192.168.21.0/24
!
management-policy CENTRALIZED-CONTROLLERS
 https server
 ssh
 user admin password 0 hellomoto role superuser access all
 no snmp-server manager v3
 snmp-server community 0 public ro
 snmp-server user snmptrap v3 encrypted des auth md5 0 Zebra
 snmp-server user snmpmanager v3 encrypted des auth md5 0 Zebra
!
management-policy CLIENT-BRIDGES
 no http server
 ssh
 user admin password 0 hellomoto role superuser access all
 no snmp-server manager v3
!
l2tpv3 policy default
!
profile nx9000 DATACENTER-NX9500
 ip name-server 192.168.10.6
 ip domain-name tmelabs.local
 no autoinstall configuration
```

```
 no autoinstall firmware
 crypto ikev1 policy ikev1-default
  isakmp-proposal default encryption aes-256 group 2 hash sha
 crypto ikev2 policy ikev2-default
  isakmp-proposal default encryption aes-256 group 2 hash sha
 crypto ipsec transform-set default esp-aes-256 esp-sha-hmac
 crypto ikev1 remote-vpn
 crypto ikev2 remote-vpn
 crypto auto-ipsec-secure
 crypto load-management
 crypto remote-vpn-client
 interface ge1
  description UPLINK
  ip dhcp trust
  qos trust dscp
  qos trust 802.1p
 interface xge1
  ip dhcp trust
  qos trust dscp
  qos trust 802.1p
 interface ge2
  ip dhcp trust
  qos trust dscp
  qos trust 802.1p
 interface xge2
  ip dhcp trust
  qos trust dscp
  qos trust 802.1p
 interface xge3
  ip dhcp trust
  qos trust dscp
  qos trust 802.1p
 interface xge4
  ip dhcp trust
  qos trust dscp
  qos trust 802.1p
 use management-policy CENTRALIZED-CONTROLLERS
 use firewall-policy default
 use auto-provisioning-policy DATACENTER
 ntp server 192.168.10.1
service pm sys-restart
!
profile ap6522 CB-AP6522
 mint level 1 area-id 65535
 ip name-server 192.168.10.6
 ip domain-name tmelabs.local
 autoinstall configuration
 autoinstall firmware
 crypto ikev1 policy ikev1-default
  isakmp-proposal default encryption aes-256 group 2 hash sha
 crypto ikev2 policy ikev2-default
  isakmp-proposal default encryption aes-256 group 2 hash sha
 crypto ipsec transform-set default esp-aes-256 esp-sha-hmac
 crypto ikev1 remote-vpn
```

```
 crypto ikev2 remote-vpn
 crypto auto-ipsec-secure
 crypto load-management
 crypto remote-vpn-client
 interface radio1
  shutdown
 interface radio2
  rf-mode bridge
  bridge ssid TMELABS-DOT1X
  bridge encryption-type ccmp
  bridge authentication-type eap
  bridge eap username eapuser1
  bridge eap password 0 hellomoto
  bridge eap type tls
 interface ge1
  ip dhcp trust
  qos trust dscp
  qos trust 802.1p
 interface vlan1
  ip address dhcp
  ip dhcp client request options all
 interface pppoe1
 use management-policy CLIENT-BRIDGES
 use firewall-policy default
 ntp server 192.168.10.1
 controller host 192.168.20.90 pool 1 level 2
 controller host 192.168.20.91 pool 2 level 2
 service pm sys-restart
 router ospf
!
rf-domain DATACENTER
location "Johnson City TN"
 contact kmarshall@Zebrasolutions.com
 timezone EST5EDT
 no country-code
!
rf-domain STORE201
 location "Johnson City TN"
 contact kmarshall@Zebrasolutions.com
 timezone EST5EDT
 country-code us
 controller-managed
!
!
nx9000 00-0C-29-91-84-D1
 use profile DATACENTER-NX9500
 use rf-domain DATACENTER
 hostname CBMGR-ACTIVE
 license AAP e3b53caa9ea6ec4fbb80bf6237a540ec514ddfbde1d0fa435d03bcdd648a2c6a8b682f0679a9b2fc
 ip default-gateway 192.168.20.1
 interface vlan1
  description MANAGEMENT
  ip address 192.168.20.90/24
 cluster name DATACENTER1
```

```
 cluster mode active
 cluster member ip 192.168.20.91 level 2
 cluster master-priority 254
 logging on
 logging console warnings
 logging buffered warnings
!
nx9000 00-0C-29-98-CB-CD
 use profile DATACENTER-NX9500
 use rf-domain DATACENTER
 hostname CBMGR-STANDBY
 ip default-gateway 192.168.20.1
 interface vlan1
  description MANAGEMENT
  ip address 192.168.20.91/24
 cluster name DATACENTER1
 cluster mode standby
 cluster member ip 192.168.20.90 level 2
 cluster master-priority 128
 logging on
 logging console warnings
 logging buffered warnings
!
!
end
```