Technical Whitepaper

Per User Tunneled Node

ArubaOS-Switch 16.04

aruba
a Hewlett Packard Enterprise company

Contents

OVERVIEW

The per-user tunneled node feature builds on top Aruba's per-port tunneled node. The per-port tunneled node feature allowed the switch to tunnel traffic to an Aruba controller on a per-port basis. The per-user tunneled node feature now gives the capability to tunnel traffic on a per-user client basis, tunneling traffic of a given client based on user role. The policies associated with that client could be driven through a RADIUS server such as ClearPass or by local user authentication in the switch.

Many devices that require power over Ethernet (PoE) and network access, such as security cameras, payment card readers, medical devices, do not have built in security software such as a desktop or laptop computer. These devices can pose a risk to networks with the lack security on the device. Per user tunneled node can authenticate these devices using ClearPass, and tunnel the client traffic, harnessing the firewall and policy capabilities in the Aruba mobility controller. This can provide secure access to IoT devices within the Campus wired network.

Per User Tunneled Node supports two types of controller deployments

- Standalone Controller Support
- Clustered Controller Support

Per User Tunneled Node Switch Platform Support

- 3810M
- 2930F/M
- 5400R (v3 blades only)

Switch Firmware Support

- ArubaOS-Switch 16.04 or later

Controller Firmware Support

- Standalone Controller (70xx,72xx : 8.1 or later
- Cluster Support in Controller (70xx,72xx): 8.1 or later

Terminologies

- Tunneled node (TN) Switch: The switch on which tunneled node profile is configured.
- Tunneled-node (TN) profile: Includes the set of parameters required to be set like controller IP, backup controller IP, etc. We can enable or disable the TN profile on a switch.
- Tunneled node (TN) interface: Interface on which tunneled node is configured.
- Client Device: The end-host (Desktop/Laptop/PoE Device) connected to a tunneled node port which is authenticated by credentials like Username/Password or mac authentication.
- Primary controller: Aruba controller working as a tunneled-node server.
- Back-up controller: Aruba back-up controller working as backup tunneled-node server.
- MM: Mobility Master
- MD: Managed Device
- Radius server: External radius server.

- LMS: Local Master Server
- UAC : User Anchor Controller
- S-UAC : Secondary User Anchor Controller
- SAC : Switch Anchor Controller
- S-SAC : Secondary Switch Anchor Controller

HOW IT WORKS

Once the tunneled-node server (controller) information is known on the switch and the "mode" is configured as role-based, the "per user tunneled node module" performs a handshake with the "tunneled-node-server" (controller) to determine its reachability and to discover the version information.

When reachability is confirmed, the per user tunneled node module in the switch software executes a switch "bootstrap". This is where the switch sends a "bootstrap message" to the controller, similar to an "AP Hello" between AP and controller. This bootstrap control packet contains user role information (VLAN, secondary user role, GRE key, etc…). Once the controller receives the message, it replies with an "acknowledge" message. Once acknowledged, the switch updates its local data structures with a "bucket map" and controller node list, which is used for mapping users to controllers and client load balancing. To further elaborate, the "bucket map" is a list of hashed entries that contain the mapping of user MAC addresses to controllers (clustered) that the users will be tunneled to. After the bucket map list is downloaded to the switch, a GRE "heartbeat" is then started between switch and controller creating a tunnel. A regular "heartbeat", using GRE, is exchanged with the controller, which then serves as the switch anchor controller (SAC). This is the "controller-ip" in the "tunneled-node-server" command. A secondary "heartbeat" is also established with the standby controller, acting as a secondary switch anchor controller (s-SAC).

As a user connects to a secure port, the authentication sub system on the switch send a RADIUS request to the RADIUS server, for example ClearPass Policy Manager, which authenticates the user and returns a user role attribute to the switch.

Attributes:

	Type	Name		Value
1.	Radius:Hewlett-Packard-Enterprise	HPE-User-Role	=	tn-secure

Figure 1: Example ClearPass User Role VSA

Once the attribute containing which user role the user will be placed in is received by the switch, the user role configured locally on the switch or downloaded from ClearPass is applied to the user.

Aruba utilizes the concept of a user role which contains user policy and access to the network based on the role. A user-role can contain policy, captive portal and VLAN information. As mentioned previously, when the user role, from the VSA received from the RADIUS server is applied to the user, a command to redirect traffic to a controller can be included within the user role. This is defined with the "tunneled-node-server-redirect" command. With this command, when the "per-user tunneled node feature" status is "up", the authentication sub system notifies the per-user tunneled node module, providing a secondary

role. The secondary role is the user role on the controller where policy generally will exist for tunneled users. This is where the firewall and security will be applied. This secondary-role information is an indication to the controller that it has to enforce additional policies to the user's traffic based on policy configuration associated with the secondary role and then form the tunnel.

In the case of users tunneled to a controller cluster, the "bucket map" containing the mapping between a given "bucket" of clients to the active user anchor controller(s) (UAC) and standby UAC (s-UAC) is populated in the Controller. A value obtained and based on the client's MAC address is assigned when a user is redirected to a controller. This value is then used to lookup the bucket map and the client device is then anchored to that particular controller node. This secondary-role information is an indication to the controller that it has to enforce additional policies to user's traffic based on policy configuration associated with the secondary role. After this process, the per user tunneled node module creates a tunnel to this UAC, if not already created, and forward user traffic to that UAC. If a user-role doesn't contain an attribute to redirect traffic to a controller then the switch will forward the traffic locally.

Once user tunnels are established to the user anchor controllers, a PAPI (Process Application Programming Interface)-based keepalive packet is exchanged with the controllers that have users anchored to them.

Per User Tunneled Node Flow

- Authenticate User
- Apply user role to authenticated user
- Redirect user traffic to controller
- Apply secondary user-role to user traffic on controller

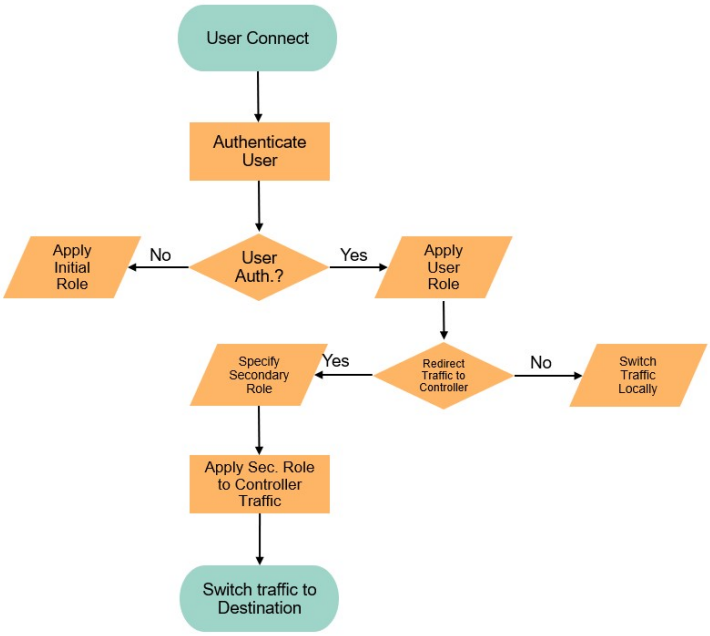

Figure 2: User Authentication Flow with per user tunneled node

CONFIGURATION

Configuring the Tunneled Node Profile on the switch:

```
(config)#tunneled-node-server
```

```
(tunneled-node-server)# controller-ip <IP-ADDR>
```

Optional: `(tunneled-node-server)# backup-controller-ip <IP-ADDR>`

Optional: `(tunneled-node-server)# keepalive interval <Integer>` - Used to change keepalive packet interval in the case of network congestion troubleshooting

```
(tunneled-node-server)# mode role-based
(tunneled-node-server)# enable
```

Note: Mode should be configured as role-based for per user tunneled node – can be changed to port-based for per port tunneled node

Configuring Local User Roles on ArubaOS-Switch

Local user roles allow user-based policy configuration local to an Aruba switch. Within the user role configuration, the command to tunnel traffic to an Aruba Mobility controller is "`tunneled-node-server-redirect`". When that command is processed, the tunnel is formed and applied to the secondary role (user role) that exists on the Mobility controller.

```
class ipv4 "testclass"
     10 match ip 0.0.0.0 255.255.255.255 0.0.0.0 255.255.255.255
     20 match tcp 0.0.0.0 255.255.255.255 0.0.0.0 255.255.255.255
   exit
policy user "testpolicy"
     10 class ipv4 "testclass" action permit
   exit
aaa authorization user-role name "tn-secure"
   policy "testpolicy"
   vlan-id 100
   tunneled-node-server-redirect secondary-role "authenticated"
   exit
aaa authorization user-role enable
```

Notes:

- The tunneled-node-server-redirect attribute instructs the switch to redirect all traffic with user-role "tn-secure" to the controller.
- The secondary-role "authenticated" specified with the redirect attribute should be configured and present on the controller.
- The switch-sent VLAN (client VLAN) has to exist on the controller.

Configuring a Tunneled Node Profile on a Mobility Controller and Cluster:

"Secondary user-role" configuration on the controller:

```
(Controller) (config) # user-role authenticated
    access-list session global-sacl
    access-list session apprf-authenticated-sacl
    access-list session ra-guard
    access-list session allowall
    access-list session v6-allowall
```

If the controller is in cluster mode the following configurations are applicable:

Mobility Master Configuration:

```
lc-cluster group-profile "hp2node"
            controller 10.0.102.6
                  controller 10.0.102.218
(ArubaMM) [mm] (config) #cd /md/00:1a:1e:02:a4:c0
(ArubaMM) [00:1a:1e:02:a4:c0] (config) #lc-cluster group-membership hp2node
(ArubaMM) [mm] (config) #cd /md/00:1a:1e:02:a6:40
(ArubaMM) [00:1a:1e:02:a6:40] (config) #lc-cluster group-membership hp2node
(ArubaMM) (config) #show  configuration node-hierarchy
      Default-node is not configured. Autopark is disabled.
      Configuration node hierarchy
      -----------------------------
   Config Node             Type
   -----------             ----
   /                       System
   /md                     System
   /md/00:1a:1e:02:a4:c0   Device
   /md/00:1a:1e:02:a6:40   Device
   /mm                     System
   /mm/mynode              System
```

Note: Configure a cluster profile. Specify the MD IP addresses. Map them to the cluster profile.

```
(ArubaMM) [mynode] (config) #show switches

All Switches
------------
IP Address      IPv6 Address  Name     Location          Type    Model      Version                Status  Configuration State     Config Sync Time (sec)  Config ID
----------      ------------  ----     --------          ----    -----      -------                ------  -------------------     ----------------------  ---------
15.212.178.108  None          ArubaMM  Building1.floor1  master  ArubaMM    8.0.0.0_55647          up      UPDATE SUCCESSFUL       0                       0
10.0.102.218    None          C2       Building1.floor1  MD      Aruba7210  8.0.0.0-hp-interop_0000 up      UNK(00:1a:1e:02:a6:40)  N/A                     N/A
10.0.102.6      None          C1       Building1.floor1  MD      Aruba7210  8.0.0.0-hp-interop_0000 up      UNK(00:1a:1e:02:a4:c0)  N/A                     N/A
```

Notes:

Verify that all nodes (MDs) are added and the status is "Update Successful"

Useful Show commands on Switch:

```
#Show tunneled-node-server state
#Show tunneled-node-server statistics
#Show tunneled-node-server information
#Show tunneled-node-server users all
#Show tunneled-node-server users count
```

Switch debug commands

```
#debug usertn
#debug destination session
#debug destination buffer
```

Useful "Show" commands for troubleshooting on Mobility controller:

```
#Show tunneled-node-mgr tunneled-nodes
#Show tunneled-node-mgr tunneled-users
#Show station-table
#Show datapath bridge
#Show datapath tunnel
#show lc-cluster group-membership
#show lc-cluster vlan-probe status
#show tunneled-node-mgr cluster-bucket-map
```

Configuring Downloadable User Roles

Downloadable user roles allow the Aruba switch to download user policy directly from ClearPass Policy Manager (CPPM). This enables ClearPass to become a centralized point in administering user policy to the access switch while also minimizing the user configuration on the switch. Downloadable user roles work by downloading user role attributes from ClearPass using the REST API. This is done using the Hyper Text Transfer Protocol Secure (HTTPS) protocol. In order for a secure Secure Sockets Layer (SSL) handshake, and for downloadable user roles to work appropriately, the signing Certificate Authority (CA) of the ClearPass HTTPS certificate must be added to the switch and marked as trusted. The following are steps in order to create the certificates necessary to configure downloadable user roles. There is a tested process that uses OpenSSL and Linux as well as one that uses Windows Server.

Creating a Root Certificate from OpenSSL in CentOS:

These steps were validated with a CentOS 7 Virtual Machine. This tested version of CentOS 7 came with a default OpenSSL version 1.0.k-fips (26 Jan 2017). These steps may vary depending on the distribution of Linux/Unix or MacOS X.

Step 1: Ensure that you have an instance of CentOS running.

Step 2: In the "Terminal" shell, change into the `/etc/pki/tls/misc` directory with the following command (may differ depending on Linux OS used):

```
cd /etc/pki/tls/misc
```
– Default path for OpenSSL

Step 3: Enter the following commands from the "misc" directory to create the certificate files and set permissions:

```
touch ../../CA/serial
chmod 777 ../../CA/serial
touch ../../CA/cacert.pem
chmod 777 ../../CA/cacert.pem
touch ../../CA/private/cakey.pem
chmod 777 ../../CA/private/cakey.pem
touch ../../CA/index.txt
chmod 777 ../../CA/index.txt
echo 1000 > /etc/pki/CA/serial
chmod 600 ../../CA/index.txt /etc/pki/CA/serial /etc/pki/tls/openssl.cnf
```

Step 4: Generate new root certificate.

```
./CA -newcert
```

Step 5: Copy `newcert.pem` (certificate generated in step 4) file into `cacert.pem`

```
cp newcert.pem ../../CA/cacert.pem
cp: overwrite `../../CA/cacert.pem'? y
```
– If existing file is already created

Step 6: Copy `newkey.pem` (private key generated in step 4) into `cakey.pem`

```
cp newkey.pem ../../CA/private/cakey.pem
cp: overwrite `../../CA/private/cakey.pem'? y
```
– If existing file is already created

Step 7: Using a text editor (vi, vim, emacs, gedit…), copy the contents of cacert.pem into a new file and save as a pem file (Ex. rootcert.pem)

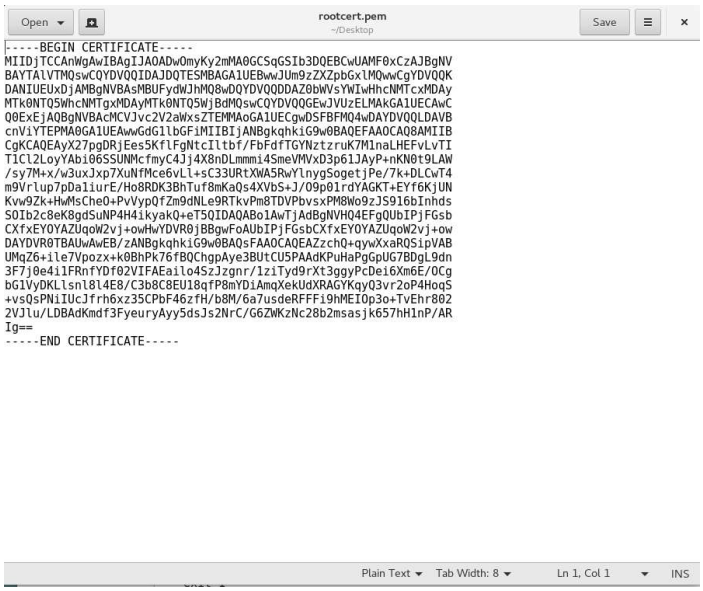

Figure 3: Gedit example: paste cacert.pem contents into new PEM file

Sign a ClearPass Certificate Signing Request using OpenSSL

Step 1: Generate Certificate Signing Request (CSR) from ClearPass Policy Manager

- Navigate to "Administration" and "Server Certificate" within ClearPass Policy Manager
- Click on "Select Type" and choose "HTTPS Server Certificate" (1.) then click on "Create Certificate Signing Request" (2.)

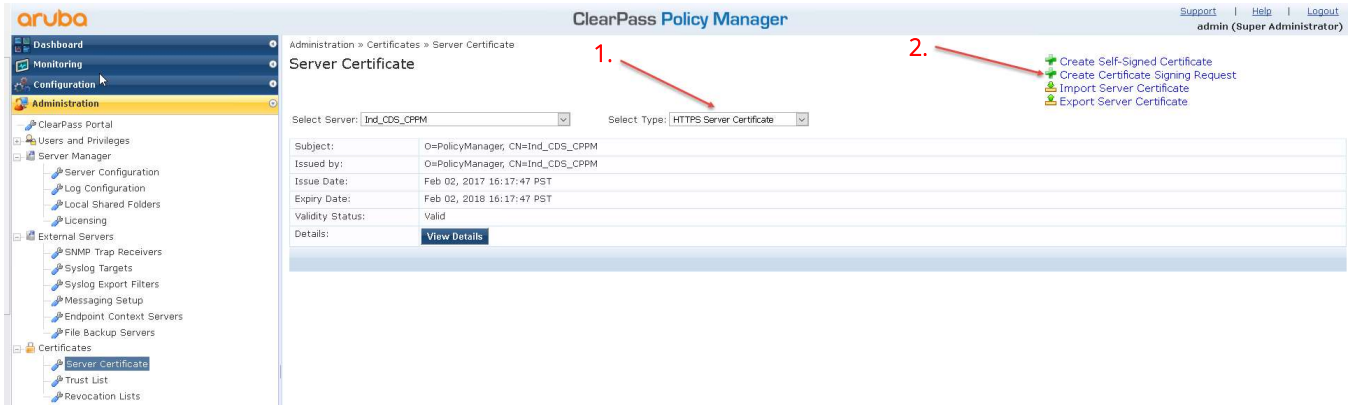

Figure 4: CPPM HTTPS Server Certificate and Certificate Signing Request creation

- Add in the appropriate information into the signing request (common name, organization, org. unit, private key, etc…)

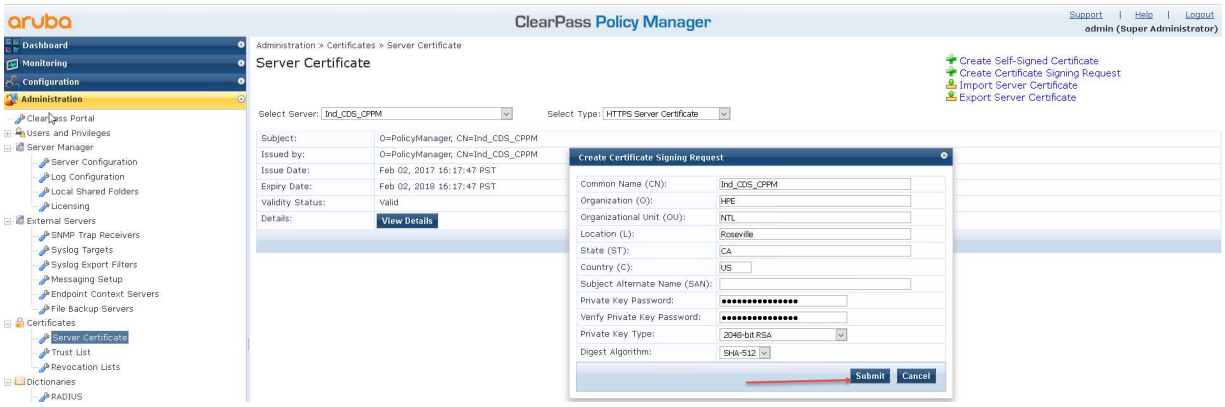

Figure 5: CPPM Create Certificate Signing Request Wizard

- Two files should then be generated by ClearPass, `CertSignRequest.csr` and `CertPrivKey.pkey` after selecting "Download CSR and Private Key File" – May depend on browser

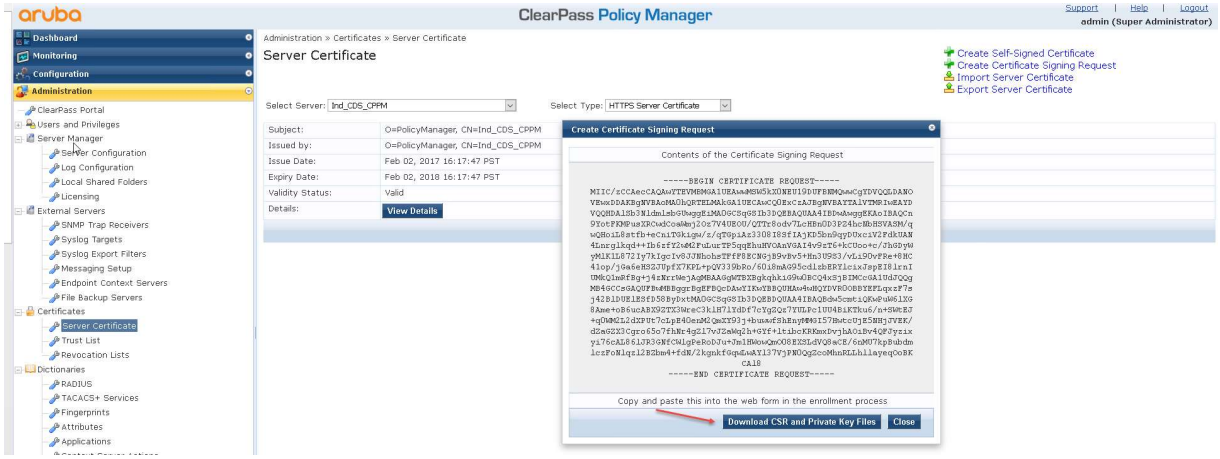

Figure 6: CPPM Certificate Signing Request Contents

Step 2: Sign ClearPass Signing Request (.csr) from OpenSSL in CentOS

- Copy the CSR file generated in Step 8 into the CentOS directory `/etc/pki/tls/misc` (may need to use SCP or FTP)

- Sign the certificate in OpenSSL using the following command:

```
sudo openssl x509 -req -days 900 -in CertSignRequest.csr -CA rootcert.pem -CAkey
../../CA/private/cakey.pem -CAcreateserial -out servercert.pem
```

Note:

  - The "`-days`" parameter will determine the expiration of the certificate
  - Make sure the root certificate created in Step 7 exists in the same directory (Ex. `/etc/pki/tls/misc`)
  - `../../CA/private/cakey.pem` refers to the private key created in Step 6
  - `servercert.pem` refers to the new HTTPS signed certificate that will be created

Step 3: Copy Root Certificate to ClearPass Trust List

Note: May need to use FTP or SCP to copy to PC/VM or ensure that ClearPass can be opened from the CentOS instance.

- Select Administration → Trust List → Add

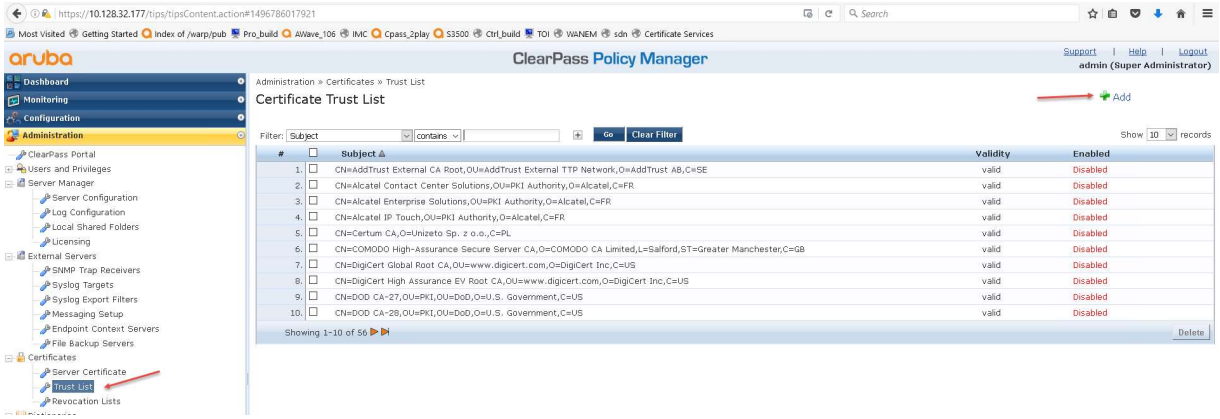

Figure 7: CPPM Trust List

- To add the root certificate to the trust list, select "Add" (1), then "Browse" (2) for the certificate to be installed and "Add Certificate" to install the root certificate to the trust list (3).

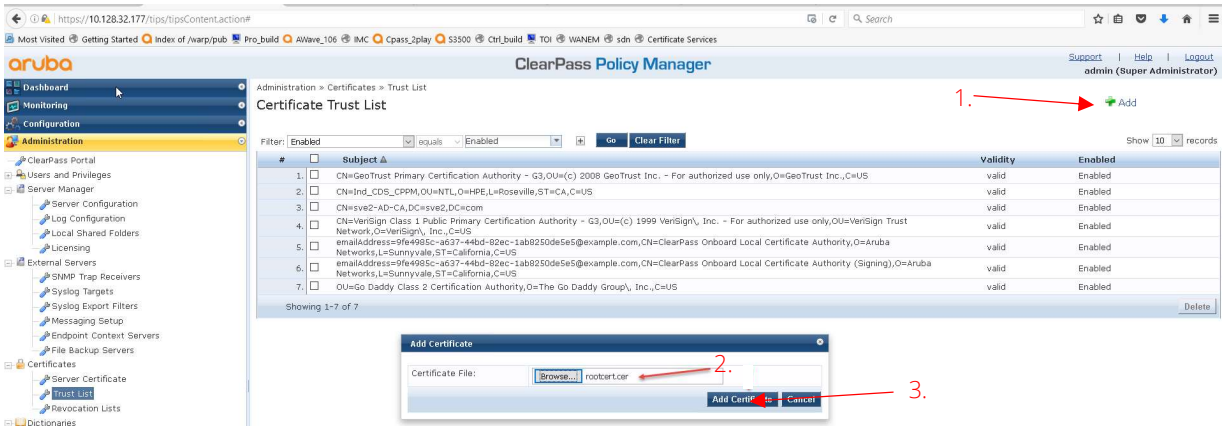

Figure 8: CPPM Add Root Certificate to Trust List

Step 4: Add HTTPS Server Certificate to ClearPass

1. Navigate to Administration → Server Certificate → Select "HTTPS Server Certificate
2. Select "Import Server Certificate"
3. Browse to server certificate created in Step 9 – May need to be copied from CentOS instance
4. Browse to Private Key File created in Step 8

5. Enter Private Key Password

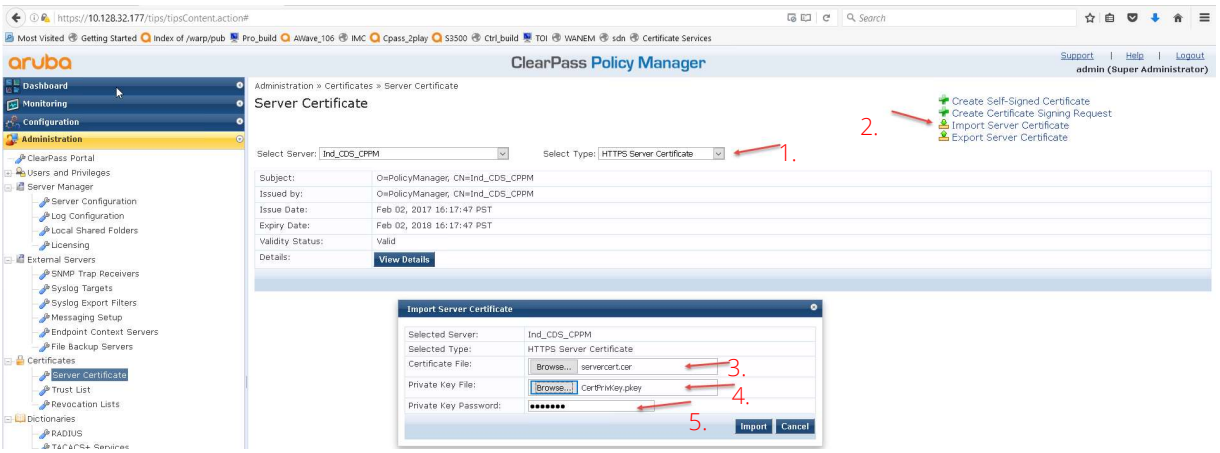

Figure 9: CPPM Add HTTPS Server Certificate

Sign ClearPass Signing Request using Windows Server

Note: In order to use this process, it is presumed that certificate services are configured and installed on Windows Server.

The following procedure was tested using Microsoft Windows Server 2016 Standard. This may work with other versions of Windows Server with variations in the steps. This process utilizes a root certificate already installed in Windows Server.

Step 1: Generate Certificate Signing Request (CSR) from ClearPass Policy Manager

- Navigate to "Administration" and "Server Certificate" within ClearPass Policy Manager

- Click on "Select Type" and choose "HTTPS Server Certificate" (1.) then click on "Create Certificate Signing Request" (2.)

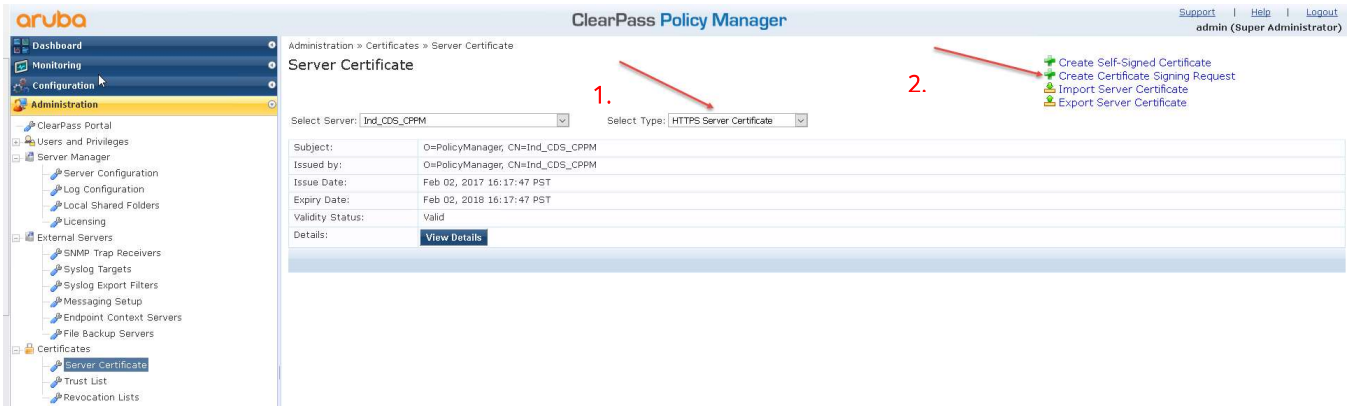

Figure 10: CPPM HTTPS Server Certificate and Certificate Signing Request creation

Step 2: Add in appropriate information into the signing request (common name, organization, org. unit, private key, etc…) and click submit

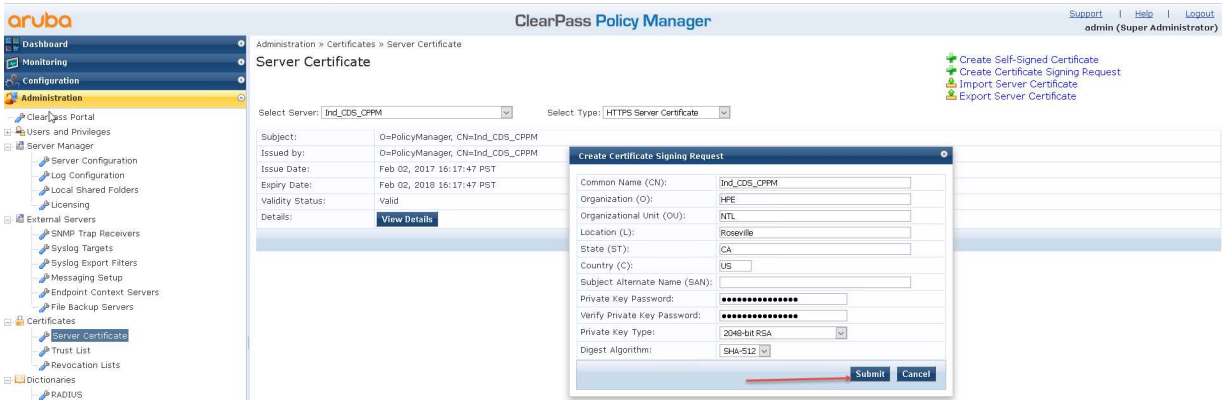

Figure 11: CPPM Certificate Signing Request Wizard

Step 3: Two files should be generated by ClearPass, `CertSignRequest.csr` and `CertPrivKey.pkey` after selecting "Download CSR and Private Key File" – May depend on browser

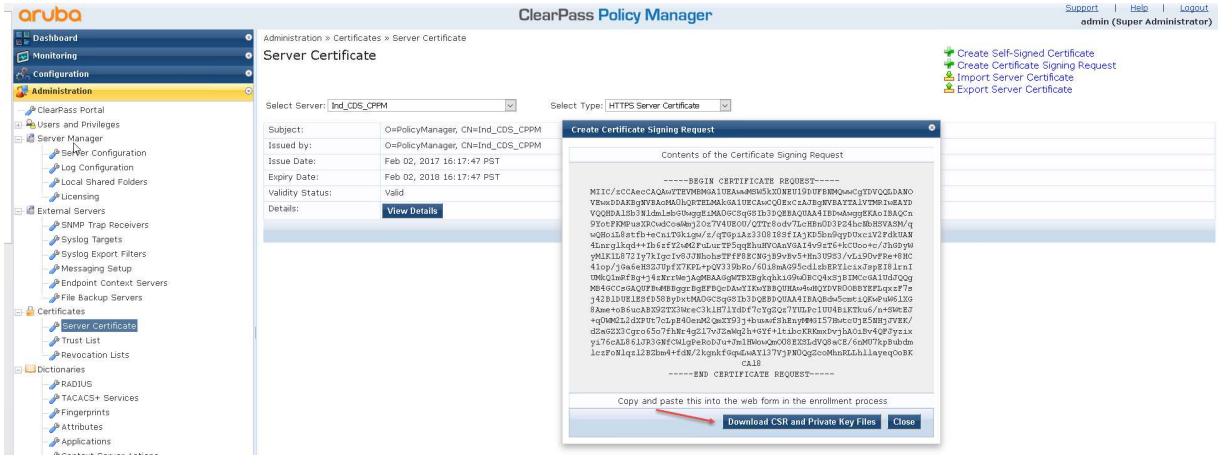

Figure 12: CPPM Certificate Signing Request Wizard

Step 4: Sign Certificate using Windows Server 2016 Standard

Open a web browser within Windows Server, navigate to https://localhost/certsrv/, then click on "Request a Certificate".

Figure 13: Windows Server Certificate Management page

Step 5: Click on "Advanced certificate request"

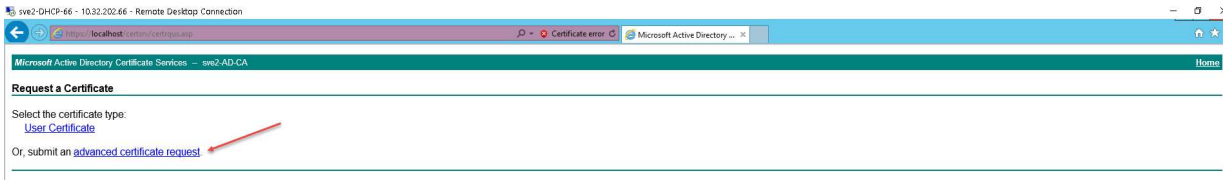

Figure 14: Windows Server advanced certificate request

Step 6: Click on "Submit a certificate request by using a base-64-encoded CMC or PKCS #10 file, or submit a renewal request by using a base-64-encoded PKCS #7 file."

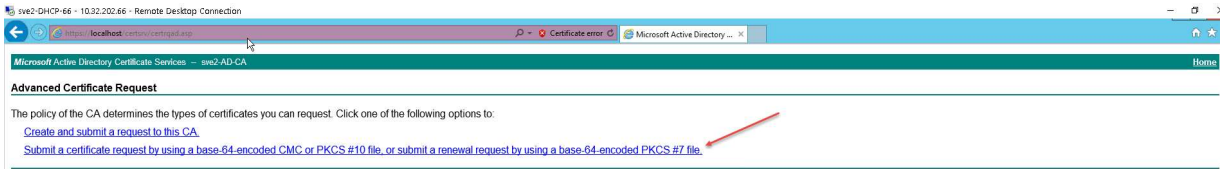

Figure 15: Windows Server submit base-64 certificate request wizard

Step 7: Copy CSR request generated by ClearPass in Step 3 and click Submit

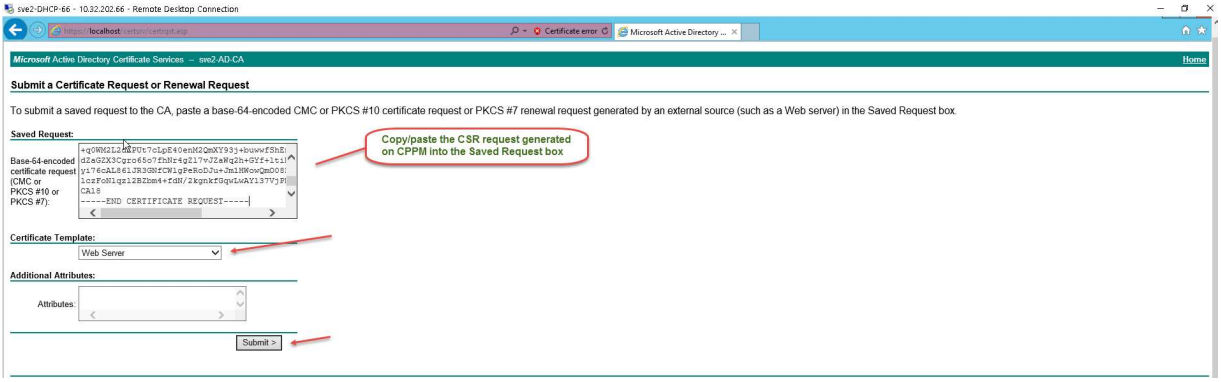

Figure 16: Windows Server copying certificate signing request (CSR) into certificate request wizard

Step 8: Click on "Base 64 encoded" and "Download certificate chain"

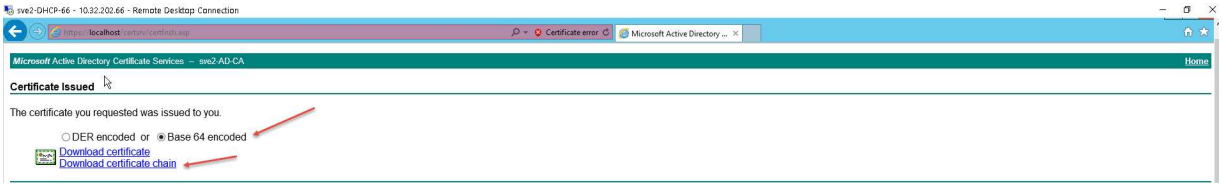

Figure 17: Windows Server complete certificate request and download certificate chain

Step 9: Certificates should open automatically in "Certificate Manager" – Both root and server certificate should appear

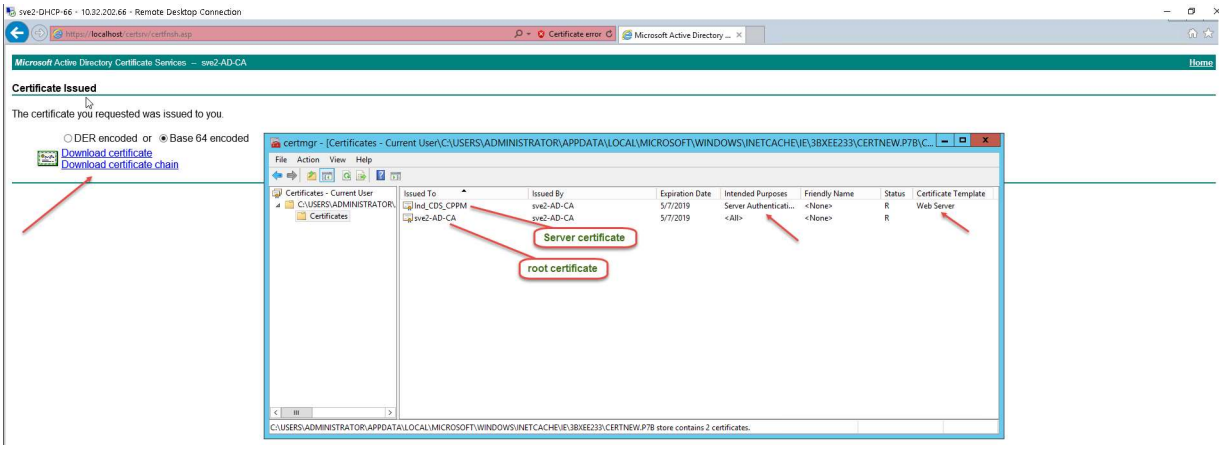

Figure 18: Windows Server certificate manager

Step 10: Right-mouse click on the server certificate (top) and left-mouse click on export, the certificate export wizard appears. Click Next

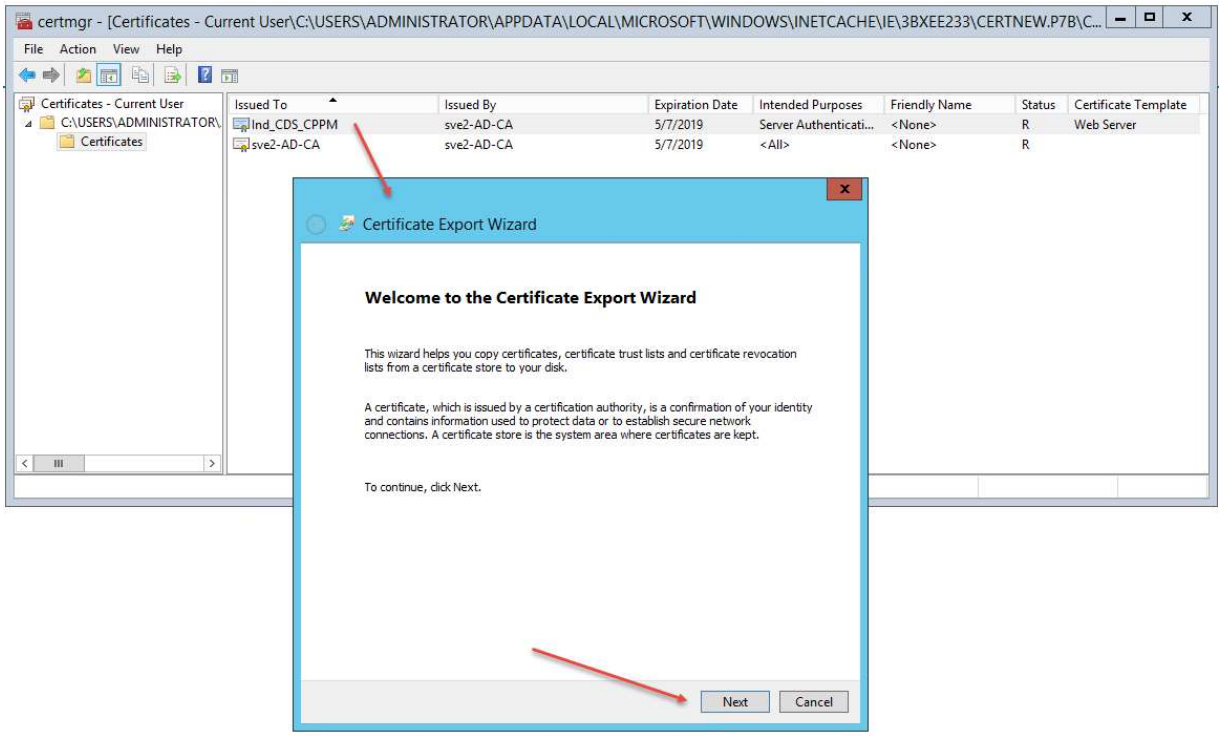

Figure 19: Windows Server certificate export wizard

Step 11: Check "Base-64 encoded X.509 (.CER) then click "Next".

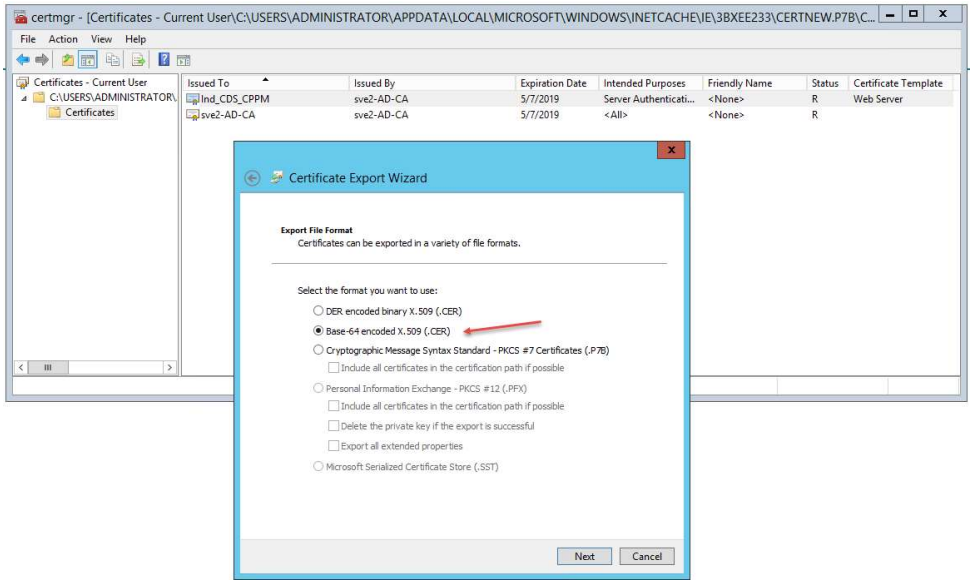

Figure 19: Windows Server certificate export wizard – Base 64 encoded selection

Step 12: Browse to the destination for the HTTPS server certificate to be stored. Click "Next".

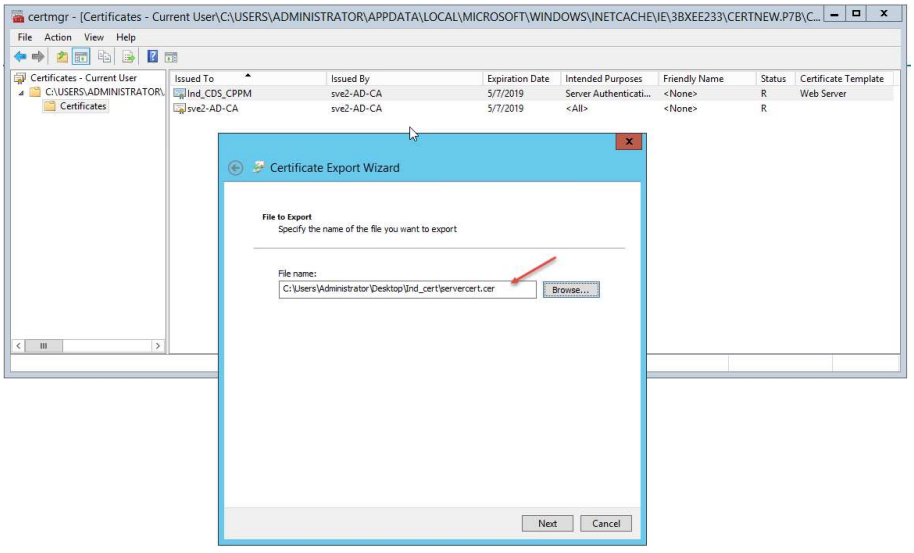

Figure 20: Windows Server certificate export wizard file destination

Step 13: Click Finish to exit the Certificate Export Wizard

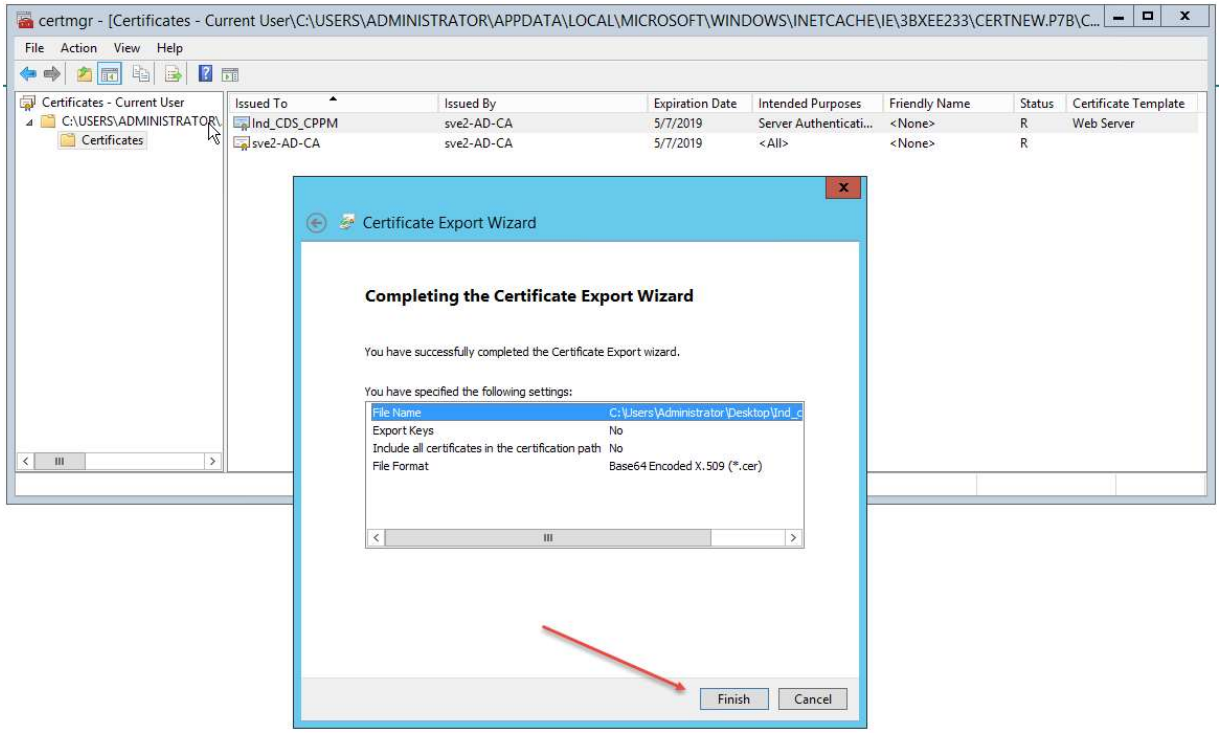

Figure 21: Windows Server certificate export wizard completion

Step 14: Repeat steps 10-13 to export the root certificate.

Step 15: Copy Root Certificate to ClearPass Trust List

Note: May need to use FTP or SCP to copy files so that they are accessible from ClearPass. Copy/Paste works with Microsoft Remote Desktop Connection.

Select Administration → Trust List → Add

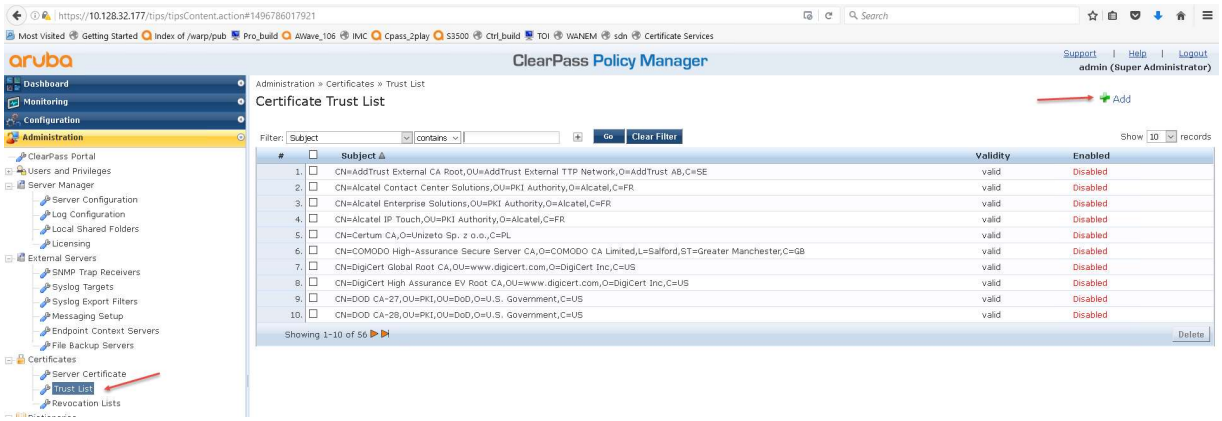

Figure 22: CPPM Add root certificate to Trust List

- To add the root certificate to the trust list, select "Add" (1), then "Browse" (2) for the certificate to be installed and "Add Certificate" to install the root certificate to the trust list (3).

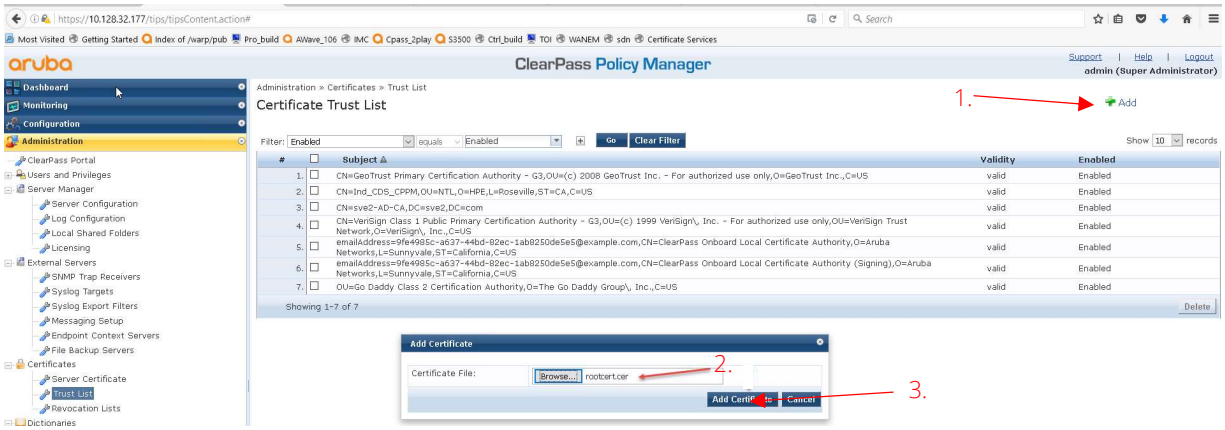

Figure 23: CPPM select root certificate location

Step 4: Add HTTPS Server Certificate to ClearPass

1. Navigate to Administration → Server Certificate → Select "HTTPS Server Certificate
2. Select "Import Server Certificate"
3. Browse to server certificate created in Steps 10-13 – May need to be copied from Windows Server instance
4. Browse to Private Key File created in Step 3

5. Enter Private Key Password

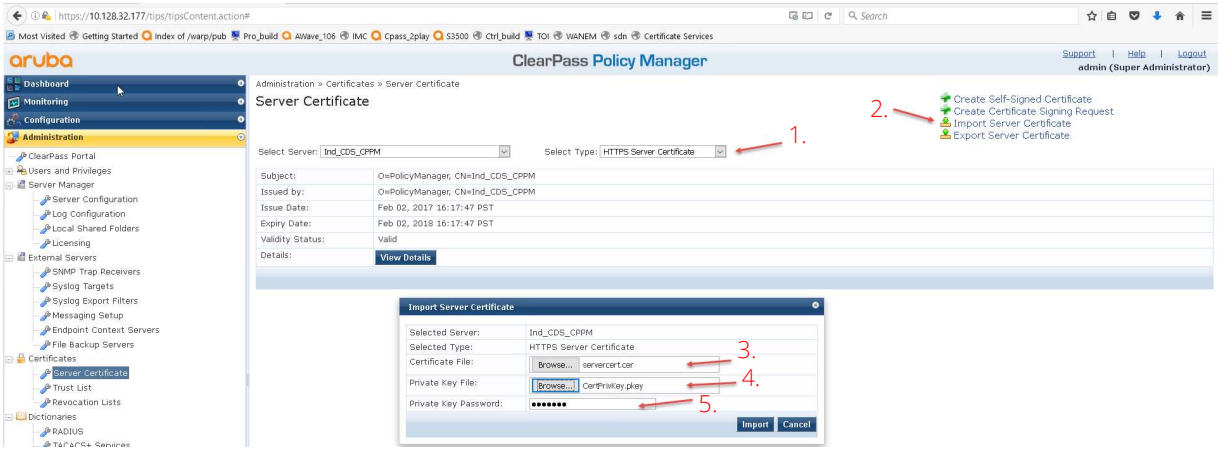

Figure 24: CPPM Add HTTPS Server Certificate and Private Key file

ClearPass Read-Only Administrator Creation

A read-only admin user is recommended to use for the switch to connect with ClearPass and download the User Role (below). These user credentials will be used within the Switch configuration to create a secure connection with ClearPass. To create a read-only admin user, navigate to Administration → Users and Privileges → Admin Users → Add.

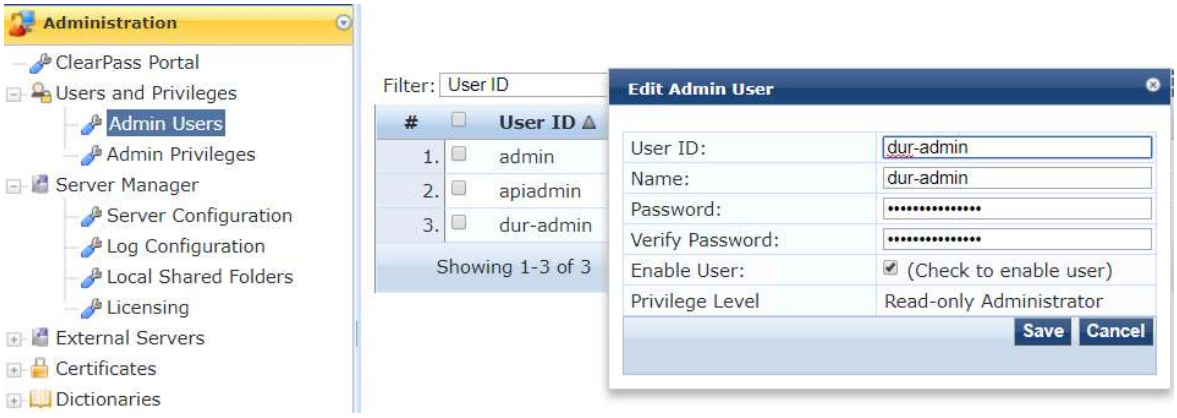

Figure 25: ClearPass Read-Only Admin User creation

Copying Root Certificate to Aruba Switch

- Create trusted anchor profile in switch where root certificate will be stored

```
(config)# crypto pki ta-profile <ta profile name>
```

Copy root certificate to switch

Note: Must use TFTP or SFTP

```
(config)# copy tftp ta-certificate <ta profile name> <TFTP server address> <root
certificate name>
```

- Verify the certificate copied correctly

```
HP-Stack-2920(config)# show crypto pki ta-profile durtest
  Profile Name    Profile Status                 CRL Configured  OCSP Configured
  --------------- ------------------------------ --------------- ---------------
  durtest         1 certificate installed         No              No

  Trust Anchor:
Version: 3 (0x2)
Serial Number:
   e0:03:c0:e9:b2:2b:2d:a6
Signature Algorithm: sha256withRSAEncryption
Issuer: C=US, ST=CA, L=Roseville, O=HPE, OU=Aruba, CN=tmelab
Validity
   Not Before: Oct  2 19:45:49 2017 GMT
   Not After : Oct  2 19:45:49 2018 GMT
Subject: C=US, ST=CA, L=Roseville, O=HPE, OU=Aruba, CN=tmelab
Subject Public Key Info:
   Public Key Algorithm: rsaEncryption
   RSA Public Key: (2048 bit)
      Modulus (2048 bit):
          30:0d:06:09:2a:86:48:86:f7:0d:01:01:01:05:00:
          03:82:01:0f:00:30:82:01:0a:02:82:01:01:00:c9:
          7d:bb:a6:00:d1:8c:47:ac:e4:a7:e5:16:03:6d:70:
          89:6d:6d:ff:c5:6c:57:5f:4c:66:0d:ce:dc:eb:b8:
          ae:cc:d6:76:8b:1c:41:6f:2e:f4:c8:4f:50:a5:d8:
          ba:32:60:06:e2:d3:a4:92:50:d3:1c:7e:6c:82:e0:
          98:f8:5f:c9:c3:2e:69:a6:8b:84:a6:79:53:15:c4:
          3d:e9:eb:52:40:c8:ff:a7:28:dd:2d:f4:b0:16:fe:
          cc:bb:33:ec:7f:c3:7b:b1:27:1a:7b:5e:e3:5f:31:
          c7:ba:bc:b9:7e:b0:2d:f7:51:1b:57:58:0e:51:c1:
          89:67:ca:04:a8:81:eb:63:3d:ef:fb:93:e0:cb:0b:
          04:f8:9b:d5:6b:96:ea:7b:a4:36:b5:8a:ea:c4:fc:
```

```
            7a:3c:44:32:b7:06:14:ee:7f:c9:8a:69:0b:38:5d:
            56:d2:f8:9f:ce:f6:9d:35:ad:d6:00:18:a4:fe:11:
            87:fa:2a:35:0d:2a:fc:3d:66:4f:87:c0:cb:02:85:
            e3:be:3e:f5:72:a5:07:d9:9b:d7:4d:2d:ef:51:4e:
            4b:cf:9b:c4:c3:54:f6:ef:b3:13:cc:f1:6a:3d:cc:
            94:bd:d7:a6:c8:9e:17:6c:48:e2:1b:d9:cf:1e:2b:
            c8:1d:4a:e3:4f:e0:7e:22:93:26:a4:43:e7:93:e5:
            02:03:01:00:01
        Exponent: 65537 (0x10001)
X509v3 extensions:
    X509v3 Subject Key Identifier:
        6c:83:e3:14:6b:1b:09:77:f1:11:83:98:01:95:2a:a1:6d:af:8f:ea
    X509v3 Authority Key Identifier:
        keyid:6c:83:e3:14:6b:1b:09:77:f1:11:83:98:01:95:2a:a1:6d:af:8f:ea
    X509v3 Basic Constraints:
        CA:TRUE

Signature Algorithm: sha256withRSAEncryption
    67:37:21:43:ea:b2:c1:7c:5a:45:04:a2:a5:50:01:50:ca:99:
    eb:e8:a5:7b:b5:69:a3:3c:7e:93:40:61:3e:4e:fa:7c:14:02:
    86:0a:40:c9:ed:c1:52:d0:94:e4:f0:00:74:a3:ee:1d:a3:e0:
    1a:95:06:ec:10:e0:2f:d7:67:dc:5e:e3:d1:ee:22:d4:54:67:
    7d:80:df:d3:65:48:14:01:1a:8a:5a:38:4b:32:73:82:7a:ff:
    d7:38:93:c9:df:6b:5e:dd:e0:83:23:dc:0d:e8:ba:5e:6e:84:
    fc:e0:a0:6c:6d:55:c8:32:8b:96:c9:e5:f2:5e:04:f3:f0:b7:
    6f:c0:bc:11:4d:7c:a9:f3:fc:99:80:e2:02:6a:97:7a:45:1d:
    5d:10:06:60:aa:b2:43:7b:eb:da:83:f8:1e:8a:92:fa:fb:10:
    b0:f3:62:21:47:09:7e:b8:7a:c7:3d:f9:08:f6:c5:e3:ac:df:
    1f:f6:fc:33:fe:9a:ee:eb:1d:79:11:45:16:2f:61:30:42:0e:
    a7:7a:3e:4e:f1:21:af:cd:36:d9:52:65:bb:f2:c3:04:07:4a:
    99:d7:f7:17:27:ae:af:20:32:cb:97:6c:26:cd:8d:ac:2f:c6:
    e9:95:8a:cc:d7:36:f1:bd:a6:b1:ab:23:93:ae:7b:84:7d:67:
    3f:f0:11:22
MD5 Fingerprint: f353 a289 81b8 72b7 3c5a da0d 8ee9 d93b
SHA1 Fingerprint: 9be9 5a27 0320 a003 1f0b 49fa 7987 4e2a 11cf 8edd
  Users associated with this TA profile
```

```
-------------------------------------
```

Configuring ArubaOS-Switch for Downloadable User Roles

Note: Ensure the Downloadable User Role VLAN exists on the switch. For example, if the downloadable user role contains VLAN 50, make sure that VLAN 50 is present in the switch configuration.

- Enabling downloadable user roles on the switch

```
aaa authorization user-role enable download
```

- Configuring switch ClearPass credentials

Note: Use Read-only Admin user credentials created in the section "ClearPass Read-Only Administrator User Creation"

```
radius-server cppm identity <user name> key <password>
```

Creating a Downloadable User Role Enforcement Profile in ClearPass

Once the credentials are created on the switch, the downloadable user role can be created in an enforcement profile within ClearPass. Referencing Figure 4, the profile will be created using the "Aruba Downloadable Role Enforcement" Category (1). Then, click the "advanced" button and "Next" to advance to configure the downloadable role (2).

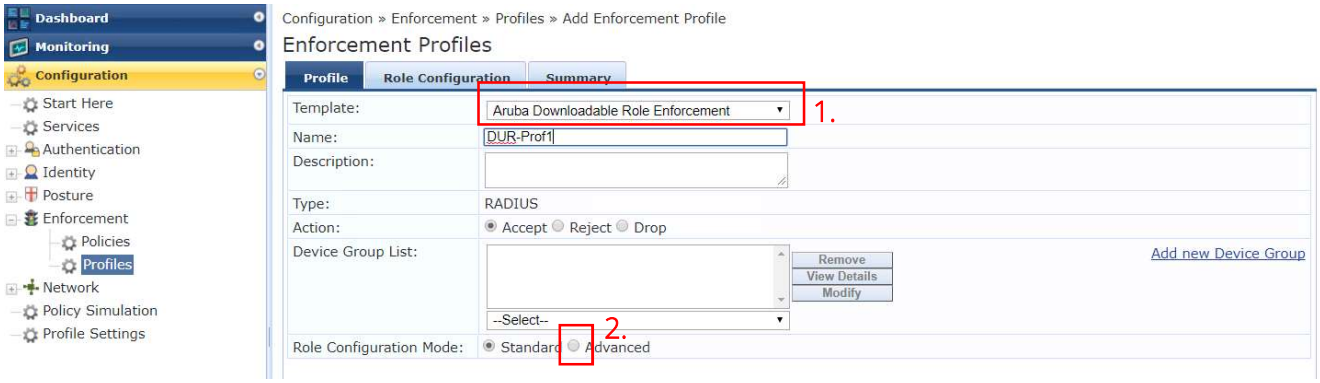

Figure 26: ClearPass Downloadable User Role Profile Creation

In figure 27, to configure the downloadable user role, use the RADIUS type “Radius: Hewlett-Packard-Enterprise” (1). Select Attribute Name “HPE-CPPM-Role” (2). In the Value field, enter the user role configuration with the exact same syntax as a local user role configuration within ArubaOS-Switch (3). Save the downloadable user role by clicking the “floppy disk” icon at the right of the attribute fields.

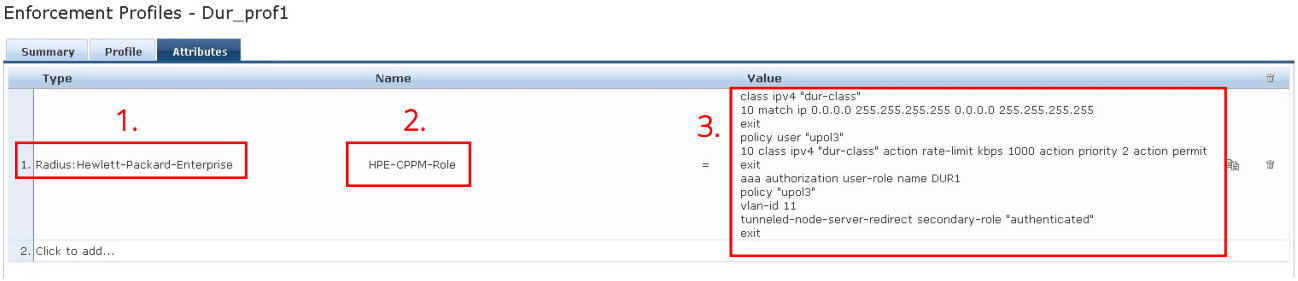

Figure 27: ClearPass Downloadable User Role Profile Creation

The downloadable user role will be configured as shown in figure 28.

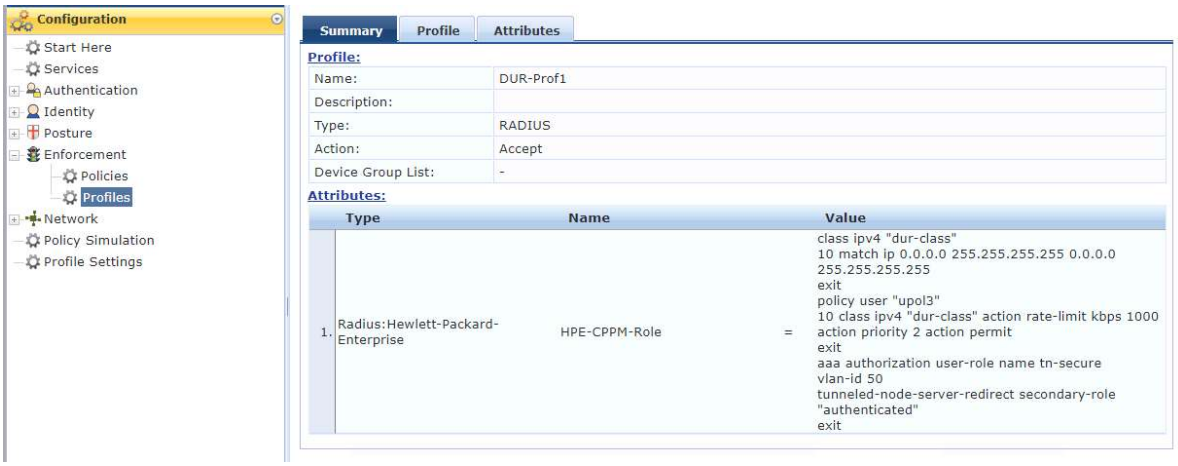

Figure 28: ClearPass Downloadable User Role Profile

SCALABILITY

Controller

Table 1 – Controller Scale Numbers

Controller	Maximum Supported Tunnels
7280	34816
7240 /7240XM	34816
7220	17408
7210	8704
7205	4352
7030	1088
7024	544
7010	544
7008	272
7005	272

Switch

Table 2 – Switch Scale Numbers

Switch or Stack	Maximum Supported User Tunnels per Switch or Stack	Maximum Supported User Tunnels per port
5400R	1024	32
3810M	1024	32
2930F	1024	32
2930M	1024	32

SUPPORTED USE CASES

Two common uses for per user tunneled node is to create wired guest capability and the ability to provide a firewall at the logical client network access device (Mobility Controller).

Wired Access Firewall

Users tunneled to an Aruba Mobility Controller can have firewall and access policy implement to restrict user access, similar to per port tunneled node. Rather than installing expensive next-generation firewalls within the network infrastructure, engineers can use the built-in firewall capabilities of the Aruba Mobility Controllers to control access.

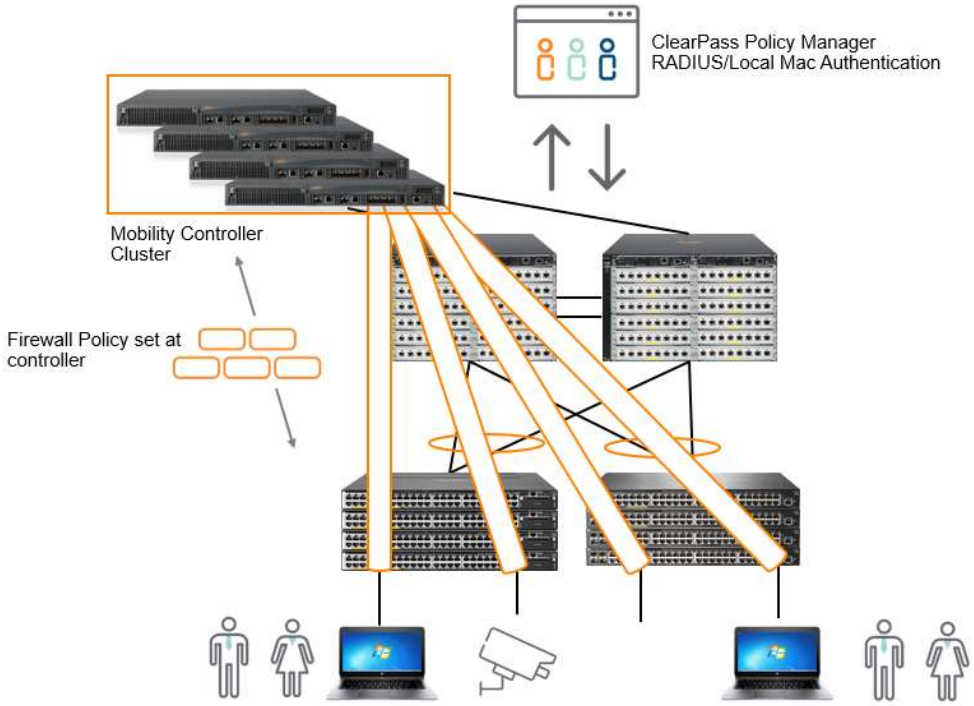

Figure 29: Wired Firewall Access Use Case Diagram

Wired Guest Segmentation

Wired guest traffic can be segmented on the network using per user tunneled node. By creating the "secondary role" on the Aruba Mobility Controller as a guest role, and assigning a specific "guest" VLAN, access and firewall policy can be implemented on the controller to isolate guest access to the rest of the campus network.

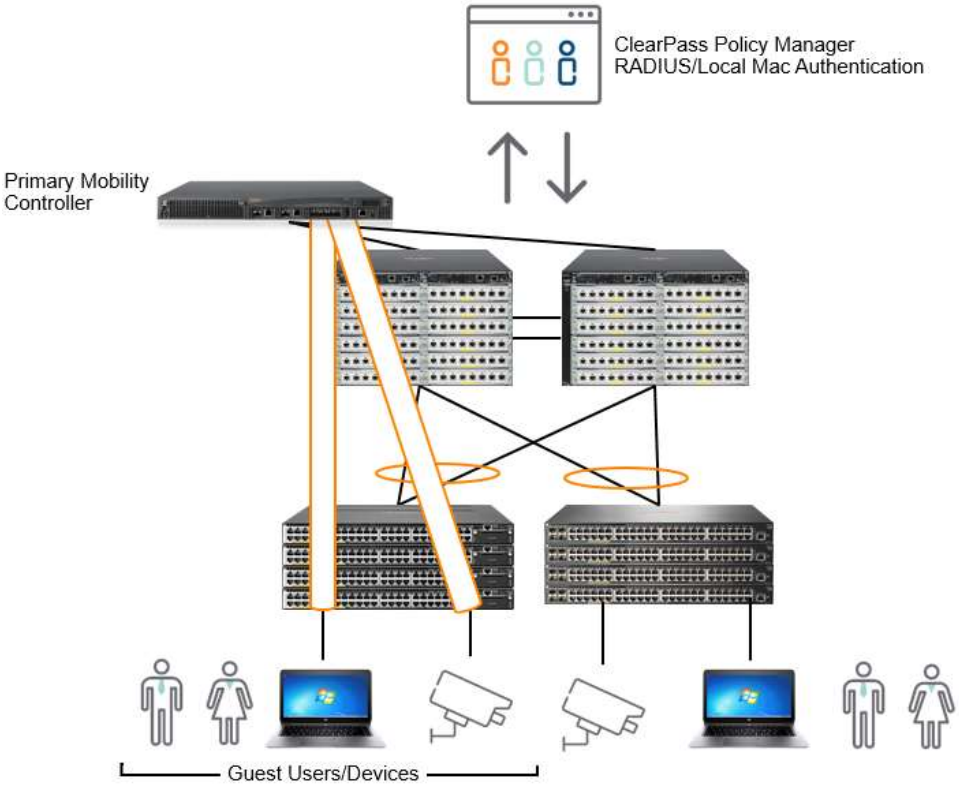

Figure 30: Wired Guest Segmentation Use Case Diagram

PARTIALLY SUPPORTED USE CASES

With the first release of per user tunneled node, some use cases will not be fully realized until future enhancements are made. The identified use cases that Aruba partially supports are: tunneled user traffic segmentation and implementing "colorless" ports (using Downloadable User Roles). Furthermore, in discussing the traffic segmentation and filtering capabilities will be dependent upon the controller as it will implement the policy for the tunneled users. This is where future enhancements will be made.

Tunneled User Traffic Segmentation

This initial release of per user tunneled node offers limited traffic segmentation in the case of the user role or device role being kept in separate VLANs. For example, if one were to tunnel IP security cameras with a user role of "camera" on VLAN 100 and Guest users were tunneled with a user role of "Guest" on VLAN 200, policy can be created restricting access to the tunneled camera traffic. The limitation comes in when trying to isolate single user or devices within the user role. One of the

fundamental design aspects to per user tunneled node, is that the tunneled VLAN must exist on the switch as well as the controller, this is carried over from per port tunneled node. Rather than having a completely dynamic way to configure the user tunnels, VLANs must statically be configured on the Aruba switch for each user role. This is also the case with downloadable user roles. When the user role is downloaded from ClearPass Policy Manager, even though the VLAN ID is identified within the downloadable user role configuration in the ClearPass enforcement profile, the VLAN still must be manually configured on the switch. Consequently, in a medium to large deployment, VLANs may need to be created on the Aruba switches using automated scripts or over the REST interface. Additionally, with implied traffic segmentation, it is assumed that multicast and broadcast replication would be handled by the controller, however, with the current implementation of per user tunneled node, it is replicated at the switch, sending traffic to all users in the same user role. Segmentation must be achieved using separate VLANs and separate roles for users and devices.

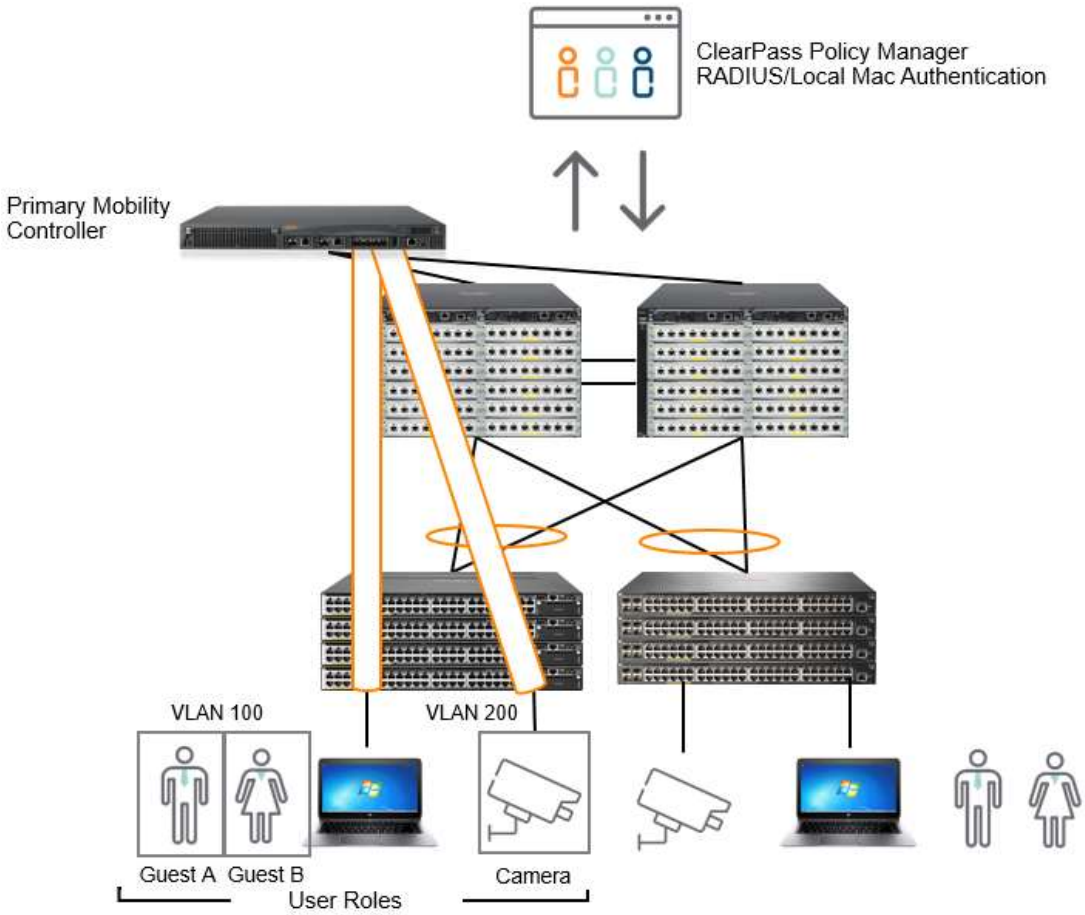

Figure 31: Traffic Segmentation Use Case Diagram

Colorless Ports

The concept of a "colorless" port at Aruba is that no matter what device or user is plugged into a switch port, or where it is plugged into a switch, it will receive the same appropriate user role and policy. Integrating with ClearPass, the new ArubaOS-Switch 16.04 release introduces Downloadable User Roles. User role and policy, previously defined on the switch (16.01 – Local User Roles), can now be downloaded from ClearPass. This, combined with per user tunneled node, is designed to give the network administrator or engineer, the most dynamically ideal way to segment and enforce policy on the network. However, as was mentioned previously, the downloadable user role VLAN must be manually configured to exist on the switch even if the user will be assigned a VLAN from the downloaded role in ClearPass. In a large- scale deployment, this could consume significant engineering time as multiple VLANs must be configured on the switch to receive the assigned client devices.

FUTURE USE CASE ENHANCEMENT

Software Defined Branch

In future enhancements to per user tunneled node, the branch use case will be enhanced. In a typical branch deployment, a switch would be connected to the controller with the wired clients and access points connected to it. A common misconfiguration would be to have the controller and switch configured to use the same VLAN. If wired clients or users were to be tunneled, there are current limitations with this design as the MAC address would "move" between the switch port and the controller port. This causes a scenario where the user tunnels start to initialize and then de-initialize. It is recommended to have the controller running in "routing" mode.

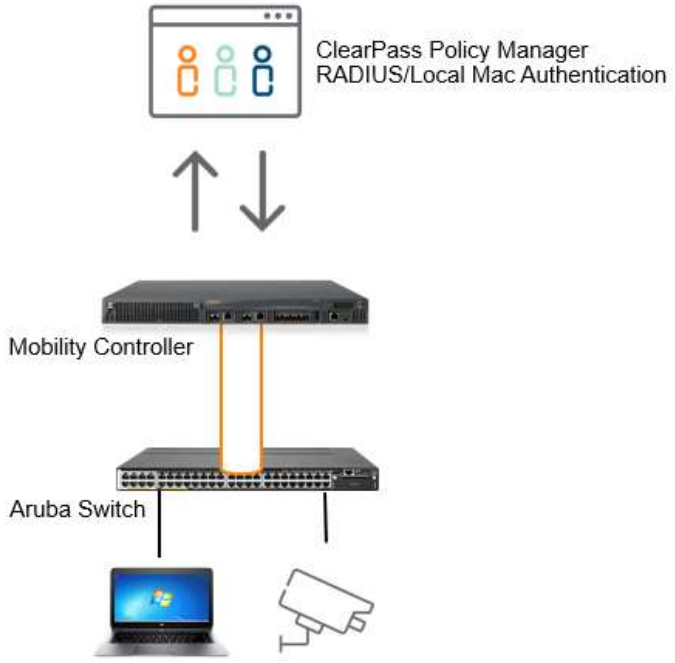

Figure 32: Branch Use Case Diagram

DEPLOYMENT SCENARIOS

Per User Tunneled Node – Standalone Controller

This deployment includes a single primary controller and an optional secondary controller which acts as back-up in the event the primary controller fails.

- On a single tunneled port, there can be as many as 32 clients with different user-roles, if clients are behind an unmanaged Layer 2 switch for example.
- On a single tunneled port, if there are two tunneled clients which are in the same role, tunneled to the same user anchor controller, a single tunnel will be formed with the controller
- On a single tunneled port, if there are two tunneled clients in different user roles, to the same user anchor controller, two tunnels will be formed with the controller
- On a single switch, if there are ten tunneled clients on ten different ports, ten tunnels are formed with the controller.

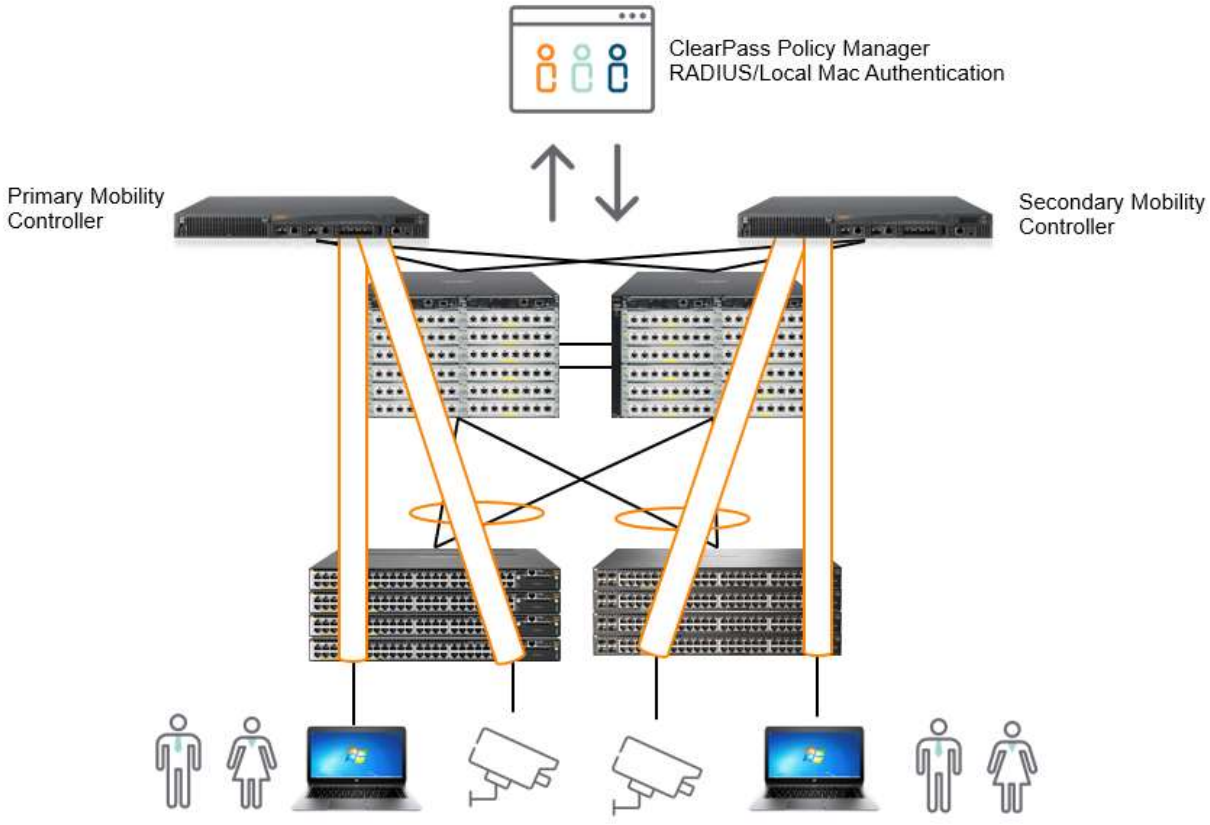

Figure 33: Standalone controller deployment diagram

Per User Tunneled Node – Controller Cluster (Wired Deployment)

This implementation utilizes the clustering of the Aruba Mobility Controller.

Controller Clustering

- The objective of clustering is to provide high availability to all the clients and ensure service continuity when a failover occurs
- The 72xx controller platform supports a maximum of 12 controllers in a cluster (All controllers = 72XX).
- The 70xx controller platform supports a maximum of 4 controllers in a cluster (All controllers = 70XX).
- If there is a mix of 70xx and 72xx controllers, a cluster can support up to a max of 4 controllers.
- In a cluster, one of the controller nodes is configured to be the Mobility Master which manages the other controller nodes which would then be called the Managed Devices.
- A cluster of controllers can be used primarily for wired tunneled traffic. This will enable a large-scale tunneled node deployment if many wired devices exist in the campus network. In the case of a large scale, wired client deployment, as long as the mobility controller resources are correctly allocated (controller clustering, load balancing, etc.), every wired port on a switch can be tunneled back to the controller.
- On a single tunneled port, if there are two tunneled clients which are in same/different roles, anchored to two different UACs, there will be a tunnel to each UAC.

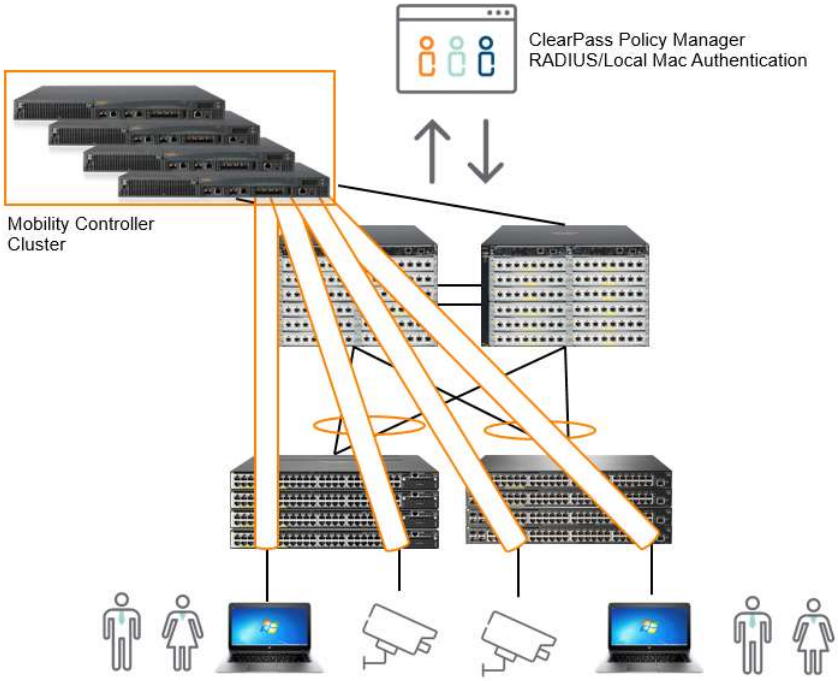

Figure 34: Clustered controller deployment diagram

Per User Tunneled Node – Controller Cluster (Large Scale - Wired and Wireless Deployment)

If there is a large scale wired and wireless deployment, a controller cluster can be used solely for per user tunneled node traffic. This will enable wired clients and devices to tunnel back to dedicated controller clusters freeing, the wireless controllers to handle pure wireless traffic.

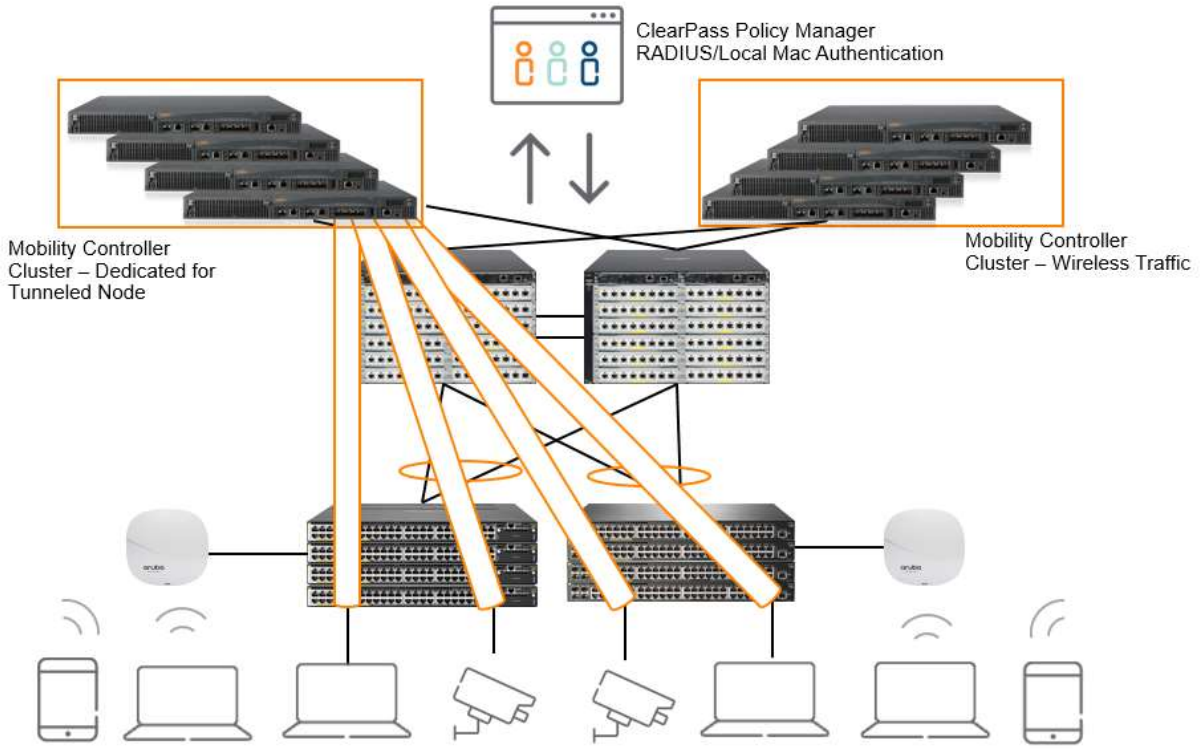

Figure 35: Wired controller cluster deployment diagram

FEATURE LIMITATIONS AND MUTUAL EXCLUSIONS

Mutually exclusive with per user tunneled node:

- Per Port Tunneled Node
- Meshing
- QinQ

Not configurable on a tunneled user port:

- Arp-protect

- DHCP-Relay
- DHCP-Server
- DHCP-Snooping
- IGMP
- MLD
- mDNS
- Openflow
- Portal commands
- SFlow
- RA-guard

FREQUENTLY ASKED QUESTIONS

1) In a controller cluster, how does the switch determine which controller to send the user traffic to?
    a. The SAC sends a bucket-map to the switch during the switch bootstrap process. This map is an array of 256 entries with each entry containing the active and standby controller to use. A user's mac address is hashed into this table to get the controller to tunnel the user traffic to.
2) When is the heartbeat started to SAC and S-SAC?
    a. Heartbeat is over a GRE tunnel with a specific GRE key (0xDEED). This is initiated with SAC and S-SAC immediately after a switch bootstrap is complete.
3) What happens when heartbeat to SAC fails?
    a. A heartbeat failure triggers the switch to:
        i. Remove users anchored to the SAC
        ii. Fail over to the S-SAC (Example: S-SAC now becomes the new SAC)
4) What happens when the keepalive to a UAC fails?
    a. The users anchored to the UAC are removed and a message is logged to the same effect in the event log.
5) Why should Jumbo frames be enabled at the switch?
    a. Jumbo frames have to be enabled at the controller uplink VLAN as well as the client VLAN. The GRE tunnel adds an effective 46 bytes to every user packet. The effective tunnel MTU = uplink VLAN MTU -46 bytes for a 1500 MTU, the tunnel MTU gets to be 1454 bytes. This means a user can send up to only 1454 bytes of frames. In order for Users to be able to send up to 1500 (default) MTU, jumbo frames need to be enabled.

6) What happens when a UAC controller goes down?

    a. A node list update is sent by the SAC to the switch to inform that a controller went down. All users anchored to that controller are removed (un-bootstrapped). After some time, the controller sends a bucket map update to the switch. The switch then processes the bucket map update and anchors users to the respective controllers (standby) as per the bucket map.

    Note: It is important to verify that the bucket map on the switch and controller are the same. Also, it should be verified that users are anchored to the right controller as shown in the bucket maps on both the switch and controller.

7) What happens when a SAC controller goes down?

    a. A node list update is sent by S-SAC to switch. Since the node list is received from the S-SAC and not the SAC, the switch considers that SAC is down and initiates a failover to S-SAC. Also, the switch removes all users anchored to SAC. Once S-SAC acknowledges the failover request, the S-SAC becomes the new active SAC. The new Active SAC then sends a node list update and bucket map update. In the node list update, the new S-SAC will be provided. The switch will then bootstrap and initiate a heartbeat with new S-SAC. The switch then processes the bucket map update and anchors users to respective controllers.

    Note: It is important to verify that the bucket map on switch and controller are the same. Also, it should be verified that users are anchored to the right controller identified in the bucket map on both the switch and controller.

8) What happens when the S-SAC controller goes down?

    a. A node list update is sent by the SAC to the switch. The switch stops the heartbeat with the S-SAC which has gone down and removes all users anchored to it. The switch then initiates a bootstrap to a new S-SAC provided in the node list update. Once a bootstrap acknowledgment is received, the switch starts a heartbeat to the new S-SAC. After some time, the SAC will send a bucket map update. The switch then processes the update and anchors users to their respective controllers.

    Note: It is important to verify that the bucket map on the switch and controller are the same. Also, it should be verified that users are anchored to the appropriate controller according to the bucket map on both the switch and controller.

9) What do the states in “show tunneled-node-server state” mean?

    a. Registering - Bootstrapping

    Registered - Bootstrapped

    Unregistering – Un-bootstrapping

10) What happens when user-role attributes change?

    a. A re-bootstrap is initiated for users applied within that role containing updated role attributes in the bootstrap packet. These users move to “registering” state. Once an acknowledgement is received from the controller, users then move to “registered” state. This applies only to VLAN and secondary role changes.

11) What happens on a client "MAC address move"?

    a. A re-bootstrap is initiated for the client. Only after an acknowledgement from the controller is received, the client traffic begins to be tunneled.

12) What is the recommendation for PUTN client VLAN configuration?

    a. Tunneled user client VLAN has to be present at the per user tunneled node switch

       There is no need to specifically add tunneled user ports to this VLAN. Switch AAA takes care of this via Mac-Based VLANs (MBV).

       The uplink to the controller port should NOT be part of this VLAN.

       The uplink to the controller VLAN and the tunneled users VLAN cannot be same.

13) "I see that user has 'registered' at the switch but has no response to a ping. How do I debug?"

    a. Check that the user roles and VLANs are correctly configured at the switch as well as the controller.

       Check the IP MTU is set to >= (1500+46) at all the switches in the path from PUTN switch to the controller.

       As there are two parts to the solution, we need to know exactly which part is not behaving right. To find out if the switch is tunneling the traffic, use the "show tunneled-node-server statistics" command to check if the user traffic is being received and transmitted. If the counters do not increment, then the switch configuration needs to be investigated.

       At the Mobility controller: check "show datapath tunnel" to see if the "Encaps" and "Decaps" counters increase.

       A packet trace of traffic sent from and received at the switch uplink to the controller can also be useful, GRE encapsulated packets are what will be of interest.

For more information

http://www.arubanetworks.com/

aruba
a Hewlett Packard Enterprise company
www.arubanetworks.com

3333 Scott Blvd | Santa Clara, CA 95054
1.866.55.ARUBA | T: 1.408.227.4500 | FAX: 1.408.227.4550 | info@arubanetworks.com