LAB GUIDE

Virtual Network-Based Tunneling (VNBT)

a Hewlett Packard Enterprise company

IMPORTANT! THIS GUIDE ASSUMES THAT THE AOS-CX OVA HAS BEEN INSTALLED AND WORKS IN GNS3 OR EVE-NG. PLEASE REFER TO GNS3/EVE-NG INITIAL SETUP LABS IF REQUIRED.

https://www.eve-ng.net/index.php/documentation/howtos/howto-add-aruba-cx-switch/

TABLE OF CONTENTS

Lab Objective

This lab will enable the reader to gain hands-on experience with Virtual Network-Based Tunneling, which uses local user roles over L2 Virtual Extensible LAN (VXLAN) and Ethernet VPN (EVPN), which is called Virtual Network-Based Tunneling at Aruba (VNBT).

Lab Overview

This lab as shown in Figure 1 will allow you to provide end hosts (Virtual PC Simulator - VPCS) on the same subnet with L2 overlay network connectivity across the VXLAN data plane tunnel created by EVPN control plane using local user roles to apply the end points into VNBT tunnels (role-based clients using VXLAN).

OSPF is used as the IP underlay Interior Gateway Protocol (IGP) to provide loopback connectivity for IBGP peering (AS#65001). IBGP EVPN with Route Reflectors (RRs) are used in this example to prevent the need for full mesh IBGP peers.

VXLAN EVPN scales better compared to flood and learn static VXLAN and allows use cases such as distributed L3 anycast gateways. Take note that L3 VXLAN does not currently work with AOS-CX VMs.

Spine1/Spine2 will function as IBGP EVPN RRs, while Leaf1/Leaf2 will function as IBGP EVPN RR clients.

VLAN 110 will be mapped to VXLAN Network Identifier (VNI) 110 to provide L2 overlay connectivity across the leaf/access switches.

Lab Network Layout

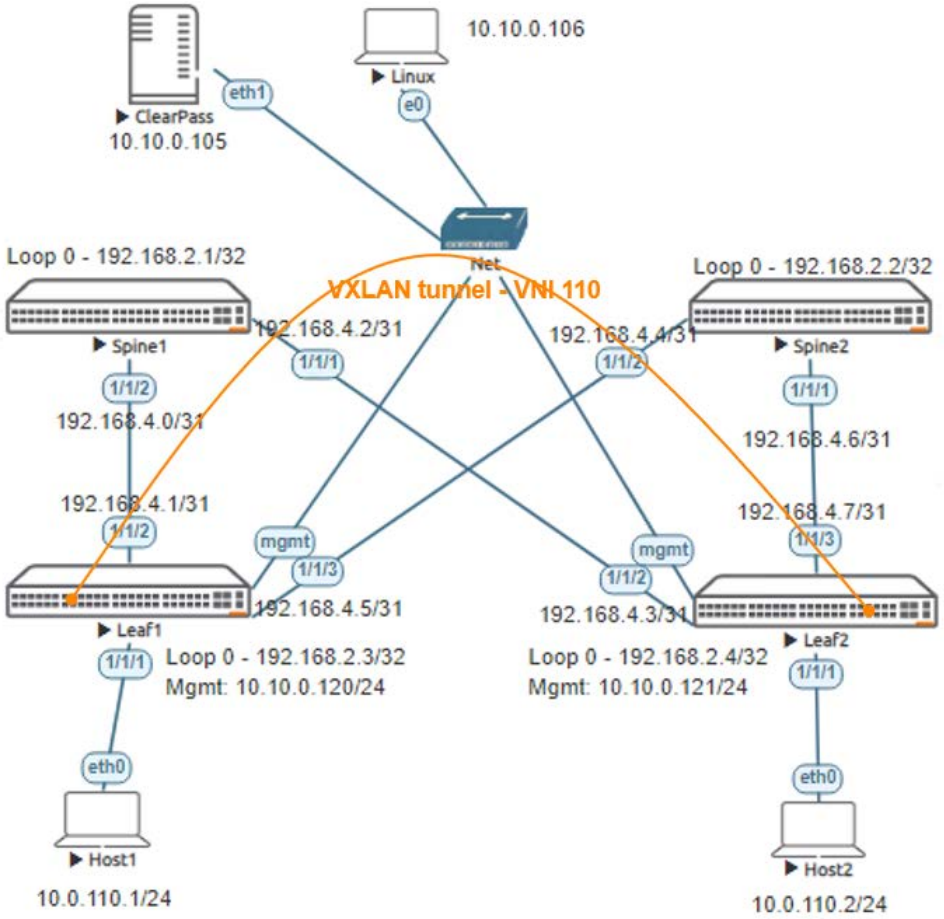


Figure 1. Lab topology and addresses

Lab Tasks

Task 1 – Lab setup

For this lab refer to Figure 1 for topology and IP address details.

- Start all the devices, including VPCS hosts
- Open each switch console and log in with user “admin” and no password
- Change all hostnames as shown in the topology:

```
configure
hostname …
```

- On all devices, bring up required ports:

```
int 1/1/1-1/1/6
```

```
  no shutdown
use “exit” to go back a level
```

- Validate LLDP neighbors appear as expected on each switch

```
show lldp neighbor
```

Leaf1

```
Leaf1(config)# sh lld neighbor-info

LLDP Neighbor Information
=========================

Total Neighbor Entries          : 2
Total Neighbor Entries Deleted  : 0
Total Neighbor Entries Dropped  : 0
Total Neighbor Entries Aged-Out : 0

LOCAL-PORT  CHASSIS-ID         PORT-ID          PORT-DESC              TTL      SYS-NAME
-------------------------------------------------------------------------------------
1/1/2       08:00:09:8a:14:fa  1/1/2            1/1/2                  120      Spine1
1/1/3       08:00:09:12:8e:9e  1/1/2            1/1/2                  120      Spine2
```

Task 2 – Configure IP Underlay Interfaces

- Configure interfaces, IPs and required VLANs on the 4 switches

Leaf1

```
Leaf1(config)# int lo 0
Leaf1(config-loopback-if)# ip add 192.168.2.3/32
Leaf1(config-loopback-if)# ip ospf 1 area 0
OSPF process does not exist.
Do you want to create (y/n)? y
OSPF Area is not configured.
Do you want to create (y/n)? y

Leaf1(config-loopback-if)# router ospf 1
Leaf1(config-ospf-1)# router-id 192.168.2.3
Leaf1(config-ospf-1)# int 1/1/2
Leaf1(config-if)# ip add 192.168.4.1/31
Leaf1(config-if)# ip ospf 1 area 0
Leaf1(config-if)# ip ospf network point-to-point
Leaf1(config-if)# int 1/1/3
Leaf1(config-if)# ip add 192.168.4.5/31
Leaf1(config-if)# ip ospf 1 area 0
Leaf1(config-if)# ip ospf network point-to-point
```

Leaf2

```
Leaf2(config)# int lo 0
Leaf2(config-loopback-if)# ip add 192.168.2.4/32
Leaf2(config-loopback-if)# ip ospf 1 area 0
OSPF process does not exist.
Do you want to create (y/n)? y
OSPF Area is not configured.
Do you want to create (y/n)? y

Leaf2(config-loopback-if)# router ospf 1
Leaf2(config-ospf-1)# router-id 192.168.2.4
```

```
Leaf2(config-ospf-1)# int 1/1/2
Leaf2(config-if)# ip add 192.168.4.3/31
Leaf2(config-if)# ip ospf 1 area 0
Leaf2(config-if)# ip ospf network point-to-point
Leaf2(config-if)# int 1/1/3
Leaf2(config-if)# ip add 192.168.4.7/31
Leaf2(config-if)# ip ospf 1 area 0
Leaf2(config-if)# ip ospf network point-to-point
```

Spine1

```
Spine1(config)# int lo 0
Spine1(config-loopback-if)# ip add 192.168.2.1/32
Spine1(config-loopback-if)# ip ospf 1 area 0
OSPF process does not exist.
Do you want to create (y/n)? y
OSPF Area is not configured.
Do you want to create (y/n)? y

Spine1(config-loopback-if)# router ospf 1
Spine1(config-ospf-1)# router-id 192.168.2.1
Spine1(config-ospf-1)# int 1/1/2
Spine1(config-if)# ip add 192.168.4.0/31
Spine1(config-if)# ip ospf 1 area 0
Spine1(config-if)# ip ospf network point-to-point
Spine1(config-if)# int 1/1/1
Spine1(config-if)# ip add 192.168.4.2/31
Spine1(config-if)# ip ospf 1 area 0
Spine1(config-if)# ip ospf network point-to-point
```

Spine2

```
Spine2(config)# int lo 0
Spine2(config-loopback-if)# ip add 192.168.2.2/32
Spine2(config-loopback-if)# ip ospf 1 area 0
OSPF process does not exist.
Do you want to create (y/n)? y
OSPF Area is not configured.
Do you want to create (y/n)? y

Spine2(config-loopback-if)# router ospf 1
Spine2(config-ospf-1)# router-id 192.168.2.2
Spine2(config-ospf-1)# int 1/1/2
Spine2(config-if)# ip add 192.168.4.4/31
Spine2(config-if)# ip ospf 1 area 0
Spine2(config-if)# ip ospf network point-to-point
Spine2(config-if)# int 1/1/1
Spine2(config-if)# ip add 192.168.4.6/31
Spine2(config-if)# ip ospf 1 area 0
Spine2(config-if)# ip ospf network point-to-point
```

- Verify OSPF neighbors appear as expected between the switches

```
Leaf1(config)# sh ip os neighbors
OSPF Process ID 1 VRF default
==============================

Total Number of Neighbors: 2

Neighbor ID      Priority  State             Nbr Address       Interface
-------------------------------------------------------------------------
192.168.2.1      n/a       FULL              192.168.4.0        1/1/2
```

```
192.168.2.2       n/a       FULL              192.168.4.4       1/1/3
```

- Verify OSPF routes are learnt as expected, you should see ECMP routes towards Lo0 of the other leaf, this is supposed to allow VXLAN traffic to be load shared across the ECMP routes (this works with real hardware, however AOS-CX VMs do not currently support ECMP)

```
Leaf1(config)# sh ip ro ospf

Displaying ipv4 routes selected for forwarding

'[x/y]' denotes [distance/metric]

192.168.2.1/32, vrf default
      via  192.168.4.0,  [110/100],  ospf
192.168.2.2/32, vrf default
      via  192.168.4.4,  [110/100],  ospf
192.168.2.4/32, vrf default                          ←ECMP to Leaf2 Lo0
      via  192.168.4.4,  [110/200],  ospf
      via  192.168.4.0,  [110/200],  ospf
192.168.4.2/31, vrf default
      via  192.168.4.0,  [110/200],  ospf
192.168.4.6/31, vrf default
      via  192.168.4.4,  [110/200],  ospf
```

Task 3 – Configure IP Underlay with EVPN

- On spine switches, configure EVPN Route Reflectors (RR) towards the leaf switches (RR clients) using leaf loopback IPs as neighbors

Spine1

```
Spine1(config)# router bgp 65001
Spine1(config-bgp)#  bgp router-id 192.168.2.1
Spine1(config-bgp)#  neighbor 192.168.2.3 remote-as 65001
Spine1(config-bgp)#  neighbor 192.168.2.3 update-source loopback 0
Spine1(config-bgp)#  neighbor 192.168.2.4 remote-as 65001
Spine1(config-bgp)#  neighbor 192.168.2.4 update-source loopback 0
Spine1(config-bgp)#  address-family l2vpn evpn
Spine1(config-bgp-l2vpn-evpn)#   neighbor 192.168.2.3 activate
Spine1(config-bgp-l2vpn-evpn)#   neighbor 192.168.2.3 route-reflector-client
BGP Session with this peer will be restarted
Spine1(config-bgp-l2vpn-evpn)#   neighbor 192.168.2.3 send-community extended
Spine1(config-bgp-l2vpn-evpn)#   neighbor 192.168.2.4 activate
Spine1(config-bgp-l2vpn-evpn)#   neighbor 192.168.2.4 route-reflector-client
BGP Session with this peer will be restarted
Spine1(config-bgp-l2vpn-evpn)#   neighbor 192.168.2.4 send-community extended
```

Spine2

```
Spine2(config-if)# router bgp 65001
Spine2(config-bgp)#  bgp router-id 192.168.2.2
Spine2(config-bgp)#  neighbor 192.168.2.3 remote-as 65001
Spine2(config-bgp)#  neighbor 192.168.2.3 update-source loopback 0
Spine2(config-bgp)#  neighbor 192.168.2.4 remote-as 65001
Spine2(config-bgp)#  neighbor 192.168.2.4 update-source loopback 0
Spine2(config-bgp)#  address-family l2vpn evpn
Spine2(config-bgp-l2vpn-evpn)#   neighbor 192.168.2.3 activate
Spine2(config-bgp-l2vpn-evpn)#   neighbor 192.168.2.3 route-reflector-client
BGP Session with this peer will be restarted
Spine2(config-bgp-l2vpn-evpn)#   neighbor 192.168.2.3 send-community extended
```

```
Spine2(config-bgp-l2vpn-evpn)#   neighbor 192.168.2.4 activate
Spine2(config-bgp-l2vpn-evpn)#   neighbor 192.168.2.4 route-reflector-client
BGP Session with this peer will be restarted
Spine2(config-bgp-l2vpn-evpn)#   neighbor 192.168.2.4 send-community extended
```

Leaf1

```
Leaf1(config)# router bgp 65001
Leaf1(config-bgp)#  bgp router-id 192.168.2.3
Leaf1(config-bgp)#  neighbor 192.168.2.1 remote-as 65001
Leaf1(config-bgp)#  neighbor 192.168.2.1 update-source loopback 0
Leaf1(config-bgp)#  neighbor 192.168.2.2 remote-as 65001
Leaf1(config-bgp)#  neighbor 192.168.2.2 update-source loopback 0
Leaf1(config-bgp)#  address-family l2vpn evpn
Leaf1(config-bgp-l2vpn-evpn)#   neighbor 192.168.2.1 activate
Leaf1(config-bgp-l2vpn-evpn)#   neighbor 192.168.2.1 send-community extended
Leaf1(config-bgp-l2vpn-evpn)#   neighbor 192.168.2.2 activate
Leaf1(config-bgp-l2vpn-evpn)#   neighbor 192.168.2.2 send-community extended
```

Leaf2

```
Leaf2(config-if)# router bgp 65001
Leaf2(config-bgp)#  bgp router-id 192.168.2.4
Leaf2(config-bgp)#  neighbor 192.168.2.1 remote-as 65001
Leaf2(config-bgp)#  neighbor 192.168.2.1 update-source loopback 0
Leaf2(config-bgp)#  neighbor 192.168.2.2 remote-as 65001
Leaf2(config-bgp)#  neighbor 192.168.2.2 update-source loopback 0
Leaf2(config-bgp)#  address-family l2vpn evpn
Leaf2(config-bgp-l2vpn-evpn)#   neighbor 192.168.2.1 activate
Leaf2(config-bgp-l2vpn-evpn)#   neighbor 192.168.2.1 send-community extended
Leaf2(config-bgp-l2vpn-evpn)#   neighbor 192.168.2.2 activate
Leaf2(config-bgp-l2vpn-evpn)#   neighbor 192.168.2.2 send-community extended
```

- Validate EVPN neighbors are up on the leaf switches

```
Leaf1(config)# show bgp l2vpn evpn summary
VRF : default
BGP Summary
-----------
 Local AS              : 65001        BGP Router Identifier  : 192.168.2.3
 Peers                 : 2            Log Neighbor Changes   : No
 Cfg. Hold Time        : 180          Cfg. Keep Alive        : 60

 Neighbor        Remote-AS MsgRcvd MsgSent   Up/Down Time State        AdminStatus
 192.168.2.1     65001       5       5       00h:01m:59s  Established   Up
 192.168.2.2     65001       5       5       00h:01m:59s  Established   Up
```

- On leaf switches, configure the desired VLAN to be VXLAN encapsulated, this VLAN will be enabled towards Host1, Host2. Specify the same vlan under evpn.
- RD and route-target can be left as auto for IBGP EVPN, these are advertised to other devices via “send-community extended” configured previously

Leaf1

```
Leaf1(config)# vlan 110
Leaf1(config-vlan-110)#
Leaf1(config-vlan-110)# evpn
Leaf1(config-evpn)#  vlan 110
```

```
Leaf1(config-evpn-vlan-110)#   rd auto
Leaf1(config-evpn-vlan-110)#   route-target export auto
Leaf1(config-evpn-vlan-110)#   route-target import auto
```

Leaf2

```
Leaf2(config)# vlan 110
Leaf2(config-vlan-110)#
Leaf2(config-vlan-110)# evpn
Leaf2(config-evpn)#  vlan 110
Leaf2(config-evpn-vlan-110)#   rd auto
Leaf2(config-evpn-vlan-110)#   route-target export auto
Leaf2(config-evpn-vlan-110)#   route-target import auto
```

Task 4 – Configure Leaf Switches with VXLAN

- Configure the VXLAN interface, the source IP based on Lo0 and the desired VLAN to VXLAN Network Identifier (VNI) mapping

Leaf1

```
Leaf1(config)# interface vxlan 1
Leaf1(config-vxlan-if)#  source ip 192.168.2.3
Leaf1(config-vxlan-if)#  no shutdown
Leaf1(config-vxlan-if)#  vni 110
Leaf1(config-vni-110)#   vlan 110
```

Leaf2

```
Leaf2(config)# interface vxlan 1
Leaf2(config-vxlan-if)#  source ip 192.168.2.4
Leaf2(config-vxlan-if)#  no shutdown
Leaf2(config-vxlan-if)#  vni 110
Leaf2(config-vni-110)#   vlan 110
```

- Validate the VXLAN interface is up with correct source, destination VTEP peer IPs via EVPN and VNI/VLAN mapping.

```
Leaf1(config)# sh int vxlan
Interface vxlan1 is up
Admin state is up
Description:
Underlay VRF: default
Destination UDP port: 4789
VTEP source IPv4 address: 192.168.2.3

VNI        VLAN   VTEP Peers        Origin
---------- ------ ----------------- --------
110        110    192.168.2.4       evpn
```

- The leafs automatically create a VXLAN tunnel between them as they are both interested in the same VNI
- Setup and start wireshark packet captures
    - right click on a leaf switch -> Capture -> 1/1/2 -> Ethernet
    - also right click on the same switch, other uplink -> Capture -> 1/1/3 -> Ethernet
- Only 1 link might show the desired packet captures as ECMP is not supported on the AOS-CX VMs

Task 5 – Configure Leaf/Access Switches for MAC Authentication

- Validate the switch has connectivity to ClearPass.

```
Switch-A# ping 10.10.0.105
PING 10.10.0.105 (10.10.0.105) 100(128) bytes of data.
108 bytes from 10.10.0.105: icmp_seq=1 ttl=64 time=1.36 ms
108 bytes from 10.10.0.105: icmp_seq=2 ttl=64 time=2.17 ms
108 bytes from 10.10.0.105: icmp_seq=3 ttl=64 time=1.17 ms
108 bytes from 10.10.0.105: icmp_seq=4 ttl=64 time=1.05 ms
108 bytes from 10.10.0.105: icmp_seq=5 ttl=64 time=1.12 ms

--- 10.10.0.105 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4004ms
rtt min/avg/max/mdev = 1.055/1.379/2.175/0.411 ms
```

- Configure the RADIUS server.
  `SwitchA(config)#radius-server host 10.10.0.105 key plaintext admin`
- From the configuration context, configure a local role on the switch using the `port-access role` command.
  ```
  Leaf1(config)#
  port-access role role1
  vlan trunk allowed 110
  ```
  Note: Ensure to add "`vlan trunk allowed 110`" to test the client connectivity.

- Configure Interface 1/1/1 on both Leaf1 and Leaf2 for MAC Authentication

Leaf1

```
Leaf1(config)# aaa authentication port-access mac-auth enable
Leaf1(config)# int 1/1/1
   aaa authentication port-access client-limit 5
   aaa authentication port-access mac-auth
        enable
```

Leaf2

```
Leaf2(config)# aaa authentication port-access mac-auth enable
Leaf2(config)# int 1/1/1
   aaa authentication port-access client-limit 5
   aaa authentication port-access mac-auth
        enable
```

Task 6 – Configure Hosts (VPCS)

- Configure Host1, Host2 with the desired IP and default gateway (the default gateway doesn't exist on the network as L2 VXLAN is used but is a required config in VPCS, so we assume a .254 as the default gateway)

Host1

```
ip 10.0.110.1/24 10.0.110.254
```

Host2

```
ip 10.0.110.2/24 10.0.110.254
```

Task 7 – ClearPass Configuration

- If running ClearPass from within the EVE-NG lab, open the Linux instance, log in using the credentials created in the Lab Setup Step 2 (default credentials - eve/eve).

Figure 2. Ubuntu Desktop in EVE-NG

- Open the Firefox Web Browser in the Linux window and navigate to 10.10.0.105.

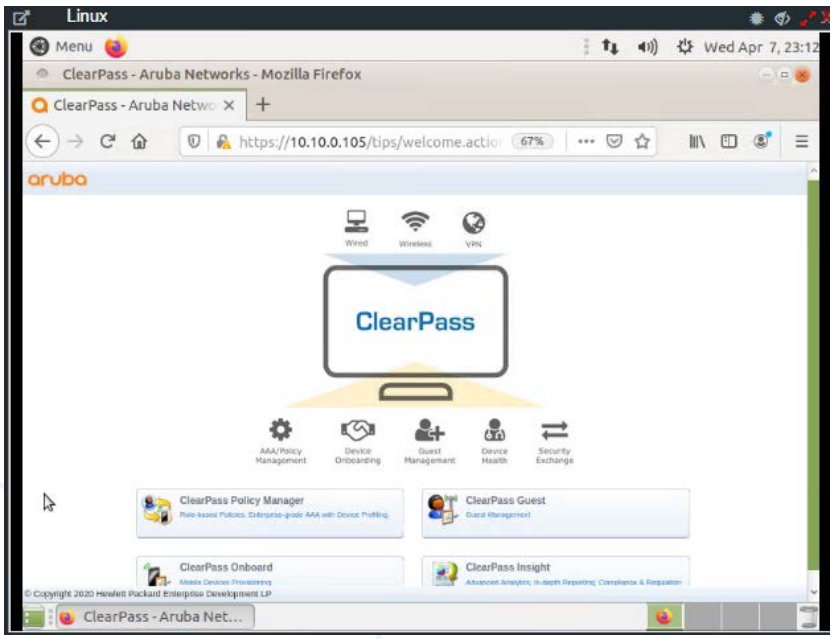


Figure 3. ClearPass Home Page in Ubuntu Window – EVE-NG

- Click on the “ClearPass Policy Manager” Button and log into ClearPass with the following credentials, ‘admin/admin123’ (or whatever password was created during setup.

Figure 4. ClearPass Login Screen

- Navigate to "Configuration → Network → Devices" and click on Devices, then click on "Add"

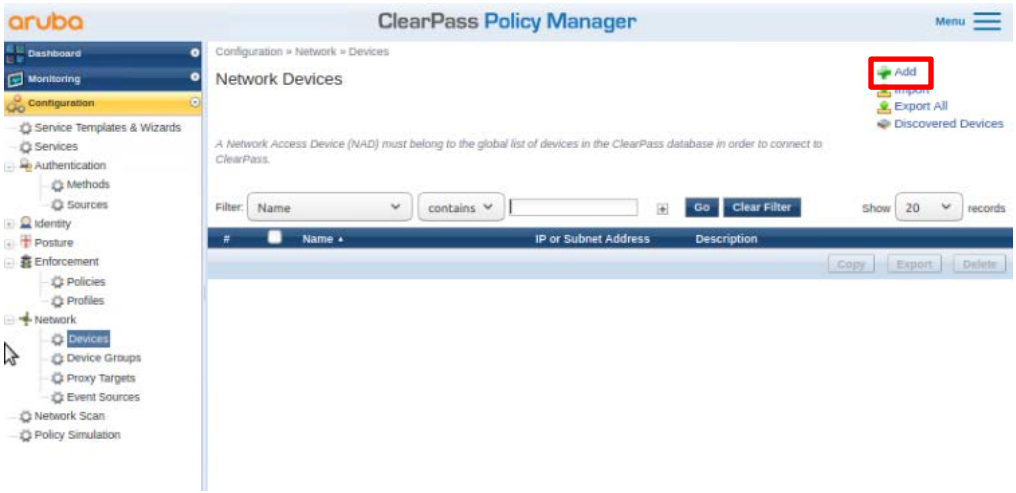


Figure 5. ClearPass Devices window

- Enter the name of the Switch that will be identified as the authenticating device in ClearPass then enter the RADIUS key and confirm it. This should match the radius-server configuration done previously on the switch.

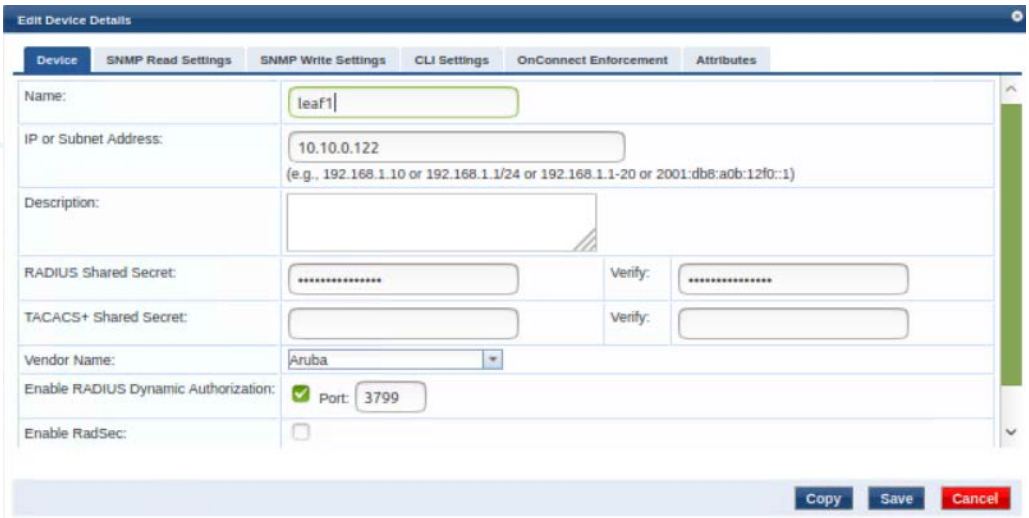


Figure 6. ClearPass Add Device Context

Note: The following steps are used to create a ClearPass Enforcement Policy for the purposes of this lab. For best practices in creating ClearPass enforcement policies in production environments, please refer to the ClearPass Policy Manager Documentation - https://www.arubanetworks.com/techdocs/ClearPass/6.9/PolicyManager/Content/home.htm. Also note that this is using MAC Authentication. 802.1x can also be used but for the purposes of this lab,

- Click on Configuration → Enforcement → Profiles → Add.

Figure 7. ClearPass Enforcement Profiles

- Select the template “Aruba RADIUS Enforcement” and give the new profile a name (Ex: ROLE1). Click Next.

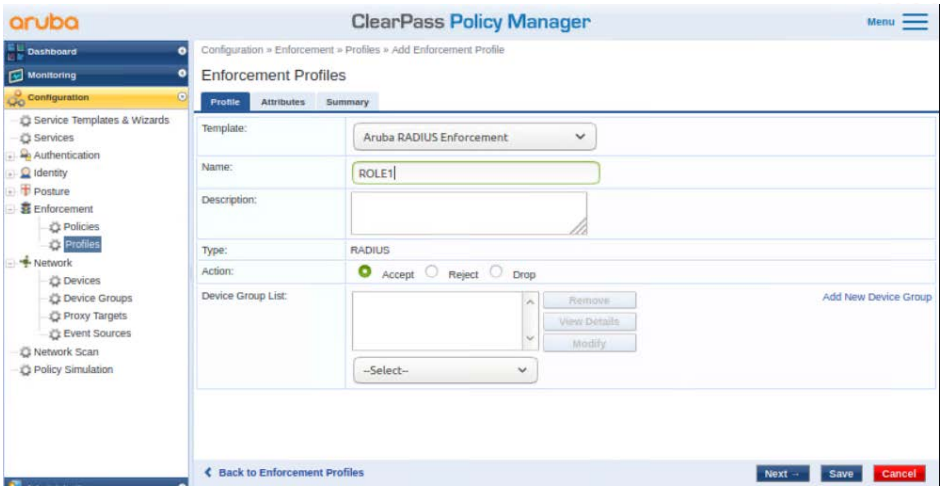


Figure 8. ClearPass Enforcement Profile creation

- Select as type “Radius:Aruba”, Name “Aruba-User-Role”, and value as the value created in the switch setup, “role1”. Click the “Save” icon (floppy disk). Click Save. Note: The role name must match the role that is configured on the switch.

Figure 9. Aruba User Role Attribute creation

- In ClearPass, click on Configuration → Services, then click on “Add”.

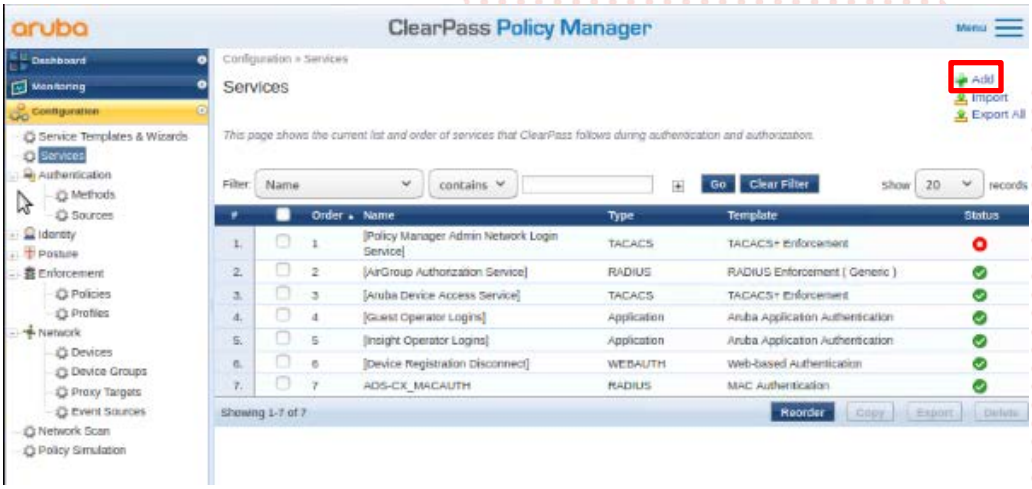


Figure 10. ClearPass Services

- Select “MAC Authentication” from the drop down and give it a name (Ex: LAB1). Click “Next”.

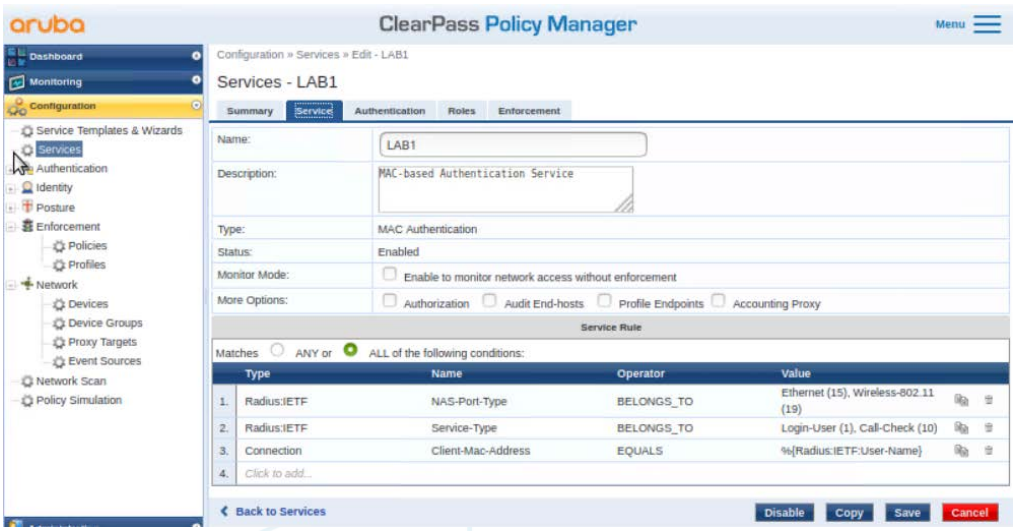


Figure 11. ClearPass MAC Authentication Service

- Select “Endpoints Repository” from the “Authentication Sources” dropdown, then click “Next”. Click “Next” again to skip the configuration of roles (not needed for this lab).

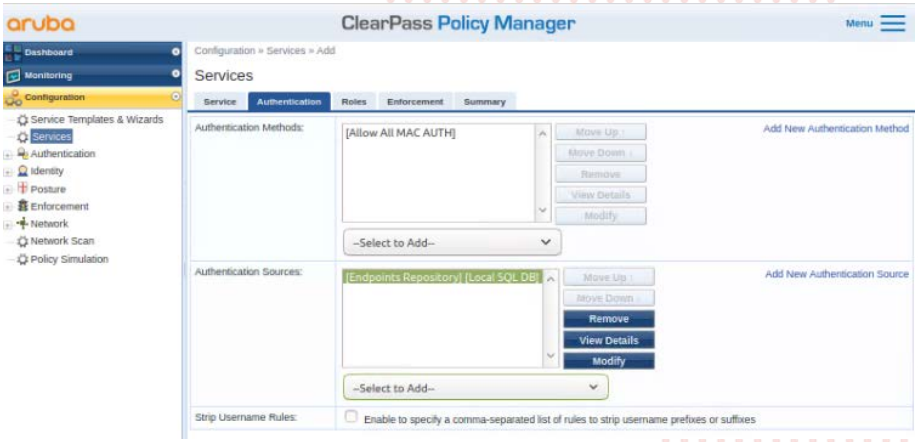


Figure 12. ClearPass MAC Authentication Sources

- From the “Enforcement” tab, click on “Add New Enforcement Policy”.

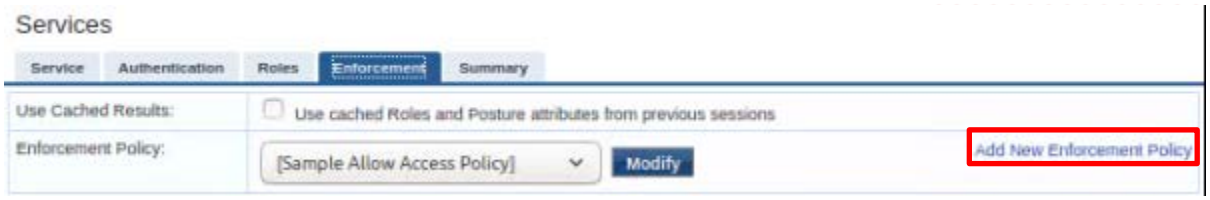


Figure 13. ClearPass Enforcement Policy

- Give the new Enforcement Policy a name (Ex: AOS-CX_ENFORCEMENT) and select “Deny Access Profile” as the default profile. Click “Next”.

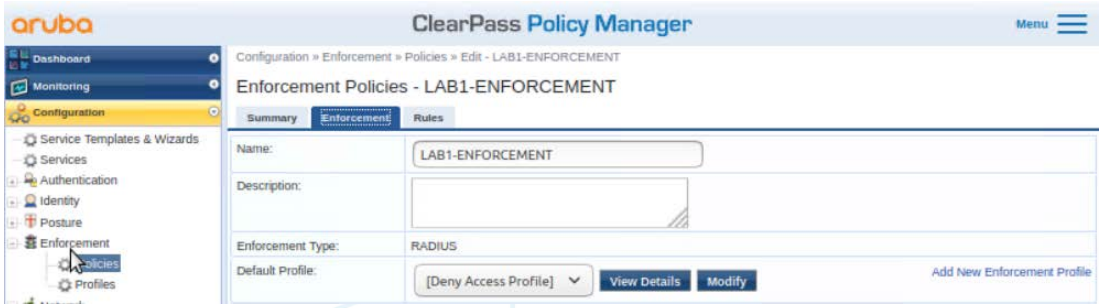


Figure 14. Adding a new Enforcement Policy

- Click on Rules and then “Add Rule”.

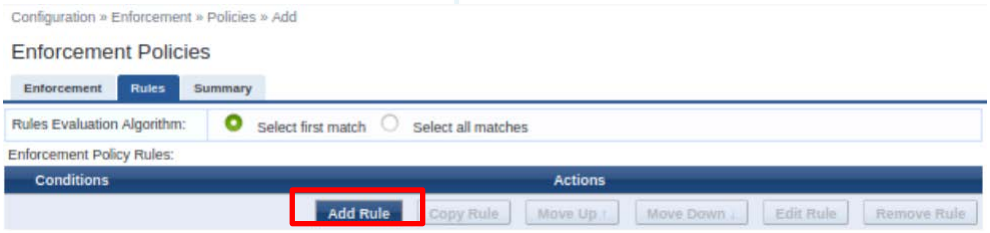


Figure 15. Adding a new Enforcement Policy

- For the purposes of this lab, we will match on the client’s MAC address, this is the MAC address that was copied from the switch configuration. Enter the Type: Connection, Name: Client-Mac-Address-Colon, Operator: BEGINS WITH, and Value

as the first 3 numbers of the client MAC Address previously retrieved from both hosts (Ex: 00:50:79). Click “Save” when finished.

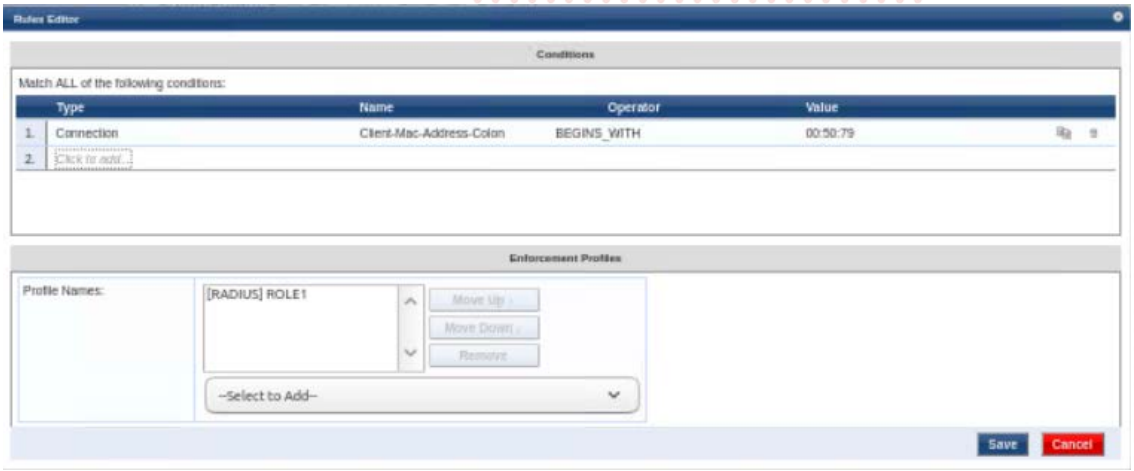


Figure 16. Adding a rule to an enforcement policy

Task 8 – Client Verification and Troubleshooting

- Open the switch console and run the command “show port-access clients”. You should see output like the following:

```
Leaf1(config)# show port-access clients

Port Access Clients

Status codes: d device-mode

--------------------------------------------------------------------------------
  Port      MAC-Address        Onboarding      Status       Role
                               Method
--------------------------------------------------------------------------------
  1/1/1     00:50:79:66:68:05 mac-auth        Success      role1
```

Note: If there is no client showing, check the access tracker in ClearPass to see if the authentication is successful. You can find that in Monitoring → Access Tracker. A successful authentication should appear as in Figure 17.

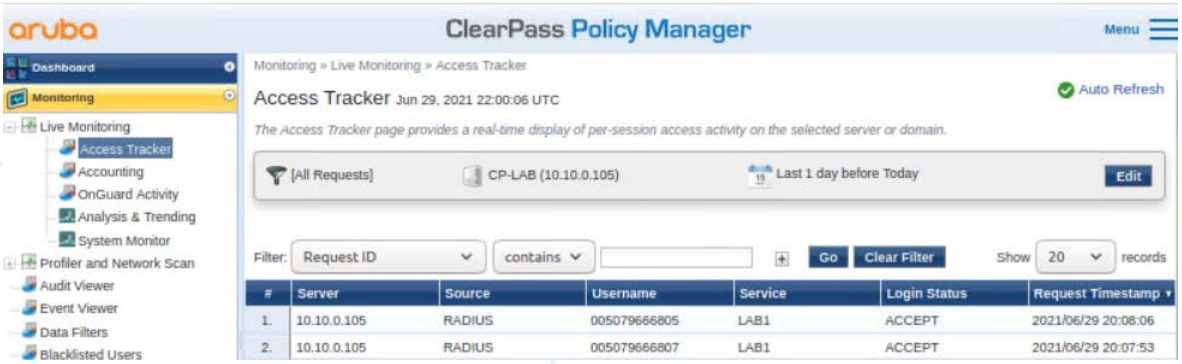


Figure 17. Successful Authentication in ClearPass Access Tracker

If the authentication were NOT successful, it would appear as a red line.

Figure 18. Unsuccessful Authentication in ClearPass Access Tracker

Click on the line and click on “Alerts” in the resulting window to see the reason why it was rejected.

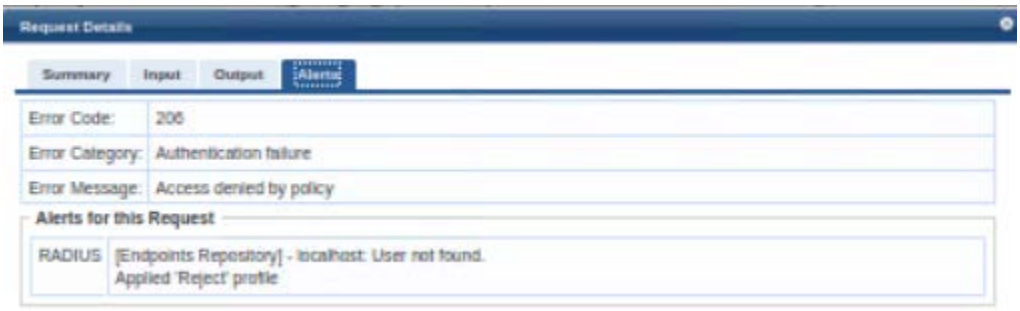


Figure 18. Unsuccessful Authentication in ClearPass Access Tracker

Also ensure that the user role name on the switch matches what is in the Aruba-User-Role attribute configured in Figure 9.

- Run the command “show port-access role local”, this gives the details of the local user role that was previously configured.

```
Leaf1(config)# show port-access role local

Role Information:

Name  : role1
Type  : local
------------------------------------------------
    Reauthentication Period             :
    Cached Reauthentication Period      :
    Authentication Mode                 :
    Session Timeout                     :
    Client Inactivity Timeout           :
    Description                         :
    Gateway Zone                        :
    UBT Gateway Role                    :
    UBT Gateway Clearpass Role          :
    Access VLAN                         :
    Native VLAN                         :
    Allowed Trunk VLANs                 : 110
    Access VLAN Name                    :
    Native VLAN Name                    :
    Allowed Trunk VLAN Names            :
    VLAN Group Name                     :
    MTU                                 :
    QOS Trust Mode                      :
    STP Administrative Edge Port        :
    PoE Priority                        :
    Captive Portal Profile              :
    Policy                              :
    GBP                                 :
```

- Run the command “show port-access clients interface 1/1/1 detail”. This gives authentication information on the interface as well as for the role that is applied to the interface.

```
Leaf1(config)# show port-access clients interface 1/1/1 detail

Port Access Client Status Details:

Client 00:50:79:66:68:05, 005079666805
==============================
  Session Details
```

```
  ---------------
    Port         : 1/1/1
    Session Time : 5309s
    IPv4 Address :
    IPv6 Address :

  VLAN Details
  ------------
    VLAN Group Name :
    VLANs Assigned  : 110--
      Access          :
      Native Untagged :
      Allowed Trunk   : 110--

  Authentication Details
  ----------------------
    Status          : mac-auth Authenticated
    Auth Precedence : dot1x - Not attempted, mac-auth - Authenticated
    Auth History    : mac-auth - Authenticated, 5211s ago
                      mac-auth - Authenticated, 5309s ago

  Authorization Details
  ---------------------
    Role   : role1
    Status : Applied


Role Information:

Name  : role1
Type  : local
-----------------------------------------------
    Reauthentication Period             :
    Cached Reauthentication Period      :
    Authentication Mode                 :
    Session Timeout                     :
    Client Inactivity Timeout           :
    Description                         :
    Gateway Zone                        :
    UBT Gateway Role                    :
    UBT Gateway Clearpass Role          :
    Access VLAN                         :
    Native VLAN                         :
    Allowed Trunk VLANs                 : 110
    Access VLAN Name                    :
    Native VLAN Name                    :
    Allowed Trunk VLAN Names            :
    VLAN Group Name                     :
    MTU                                 :
    QOS Trust Mode                      :
    STP Administrative Edge Port        :
    PoE Priority                        :
    Captive Portal Profile              :
    Policy                              :
    GBP                                 :
```

Task 9 – Final Validation

- Ensure L2 connectivity works between hosts – user roles should be applied and hosts should be placed into VLAN 110 which is attached to VNI 110 – Test connectivity through the tunnel

```
VPCS> ping 10.0.110.2

84 bytes from 10.0.110.2 icmp_seq=1 ttl=64 time=1.787 ms
84 bytes from 10.0.110.2 icmp_seq=2 ttl=64 time=3.202 ms
84 bytes from 10.0.110.2 icmp_seq=3 ttl=64 time=3.999 ms
84 bytes from 10.0.110.2 icmp_seq=4 ttl=64 time=3.055 ms
84 bytes from 10.0.110.2 icmp_seq=5 ttl=64 time=3.375 ms
```

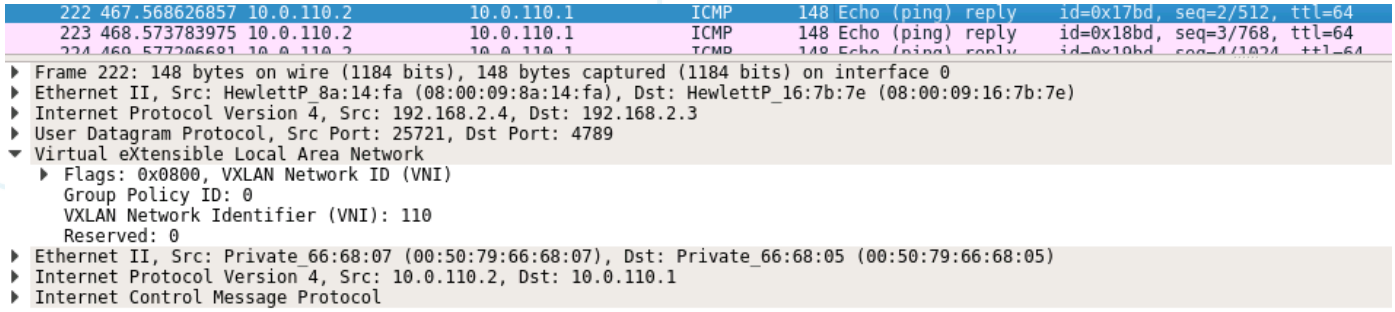
- Validate local and remote MACs are seen on the leaf switches as expected

```
Leaf1# sh mac-address-table
MAC age-time                : 300 seconds
Number of MAC addresses : 2

MAC Address          VLAN     Type                        Port
-------------------------------------------------------------
00:50:79:66:68:05    110      dynamic                     1/1/1
00:50:79:66:68:07    110      evpn                        vxlan1(192.168.2.4)
```

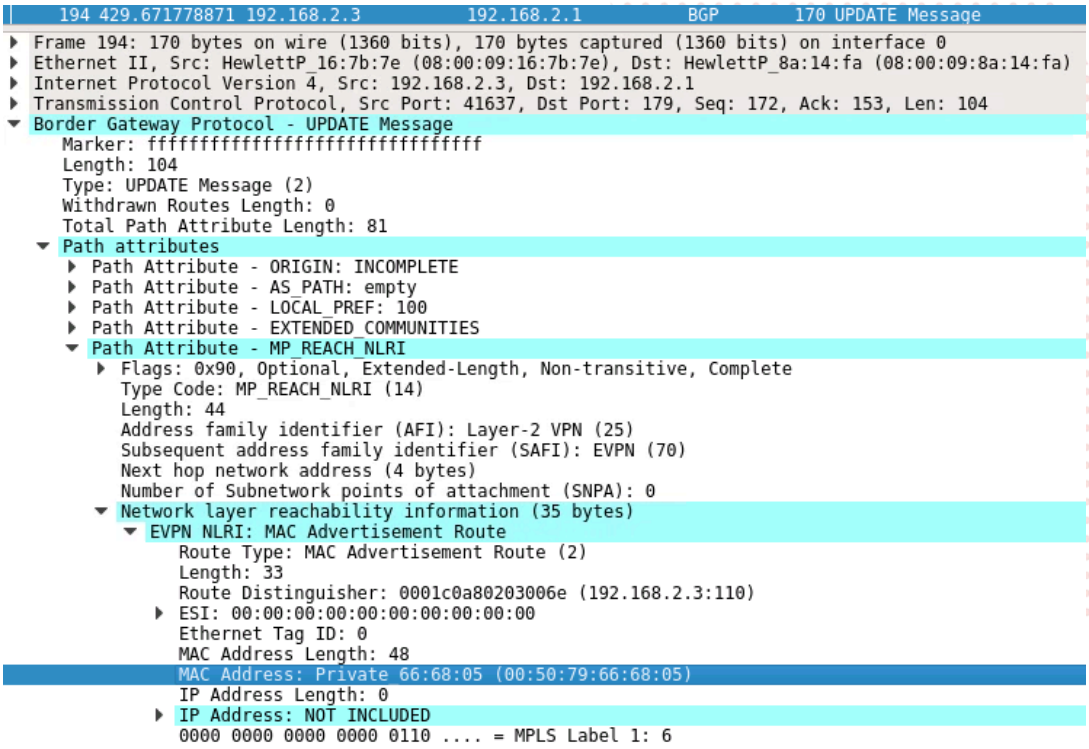
- Validate local and remote MACs are also seen in the EVPN table

```
Leaf1# sh bgp l2vpn evpn
Status codes: s suppressed, d damped, h history, * valid, > best, = multipath,
              i internal, e external S Stale, R Removed
Origin codes: i - IGP, e - EGP, ? - incomplete

EVPN Route-Type 2 prefix: [2]:[ESI]:[EthTag]:[MAC]:[OrigIP]
EVPN Route-Type 3 prefix: [3]:[EthTag]:[OrigIP]
VRF : default
Local Router-ID 192.168.2.3

     Network                                                Nexthop         Metric     LocPrf     Weight
Path
-------------------------------------------------------------------------------------------------------------------
-
Route Distinguisher: 192.168.2.3:110      (L2VNI 110)
*>  [2]:[0]:[0]:[00:50:79:66:68:05]:[]                     192.168.2.3     0          100        0       ?
*>  [3]:[0]:[192.168.2.3]                                  192.168.2.3     0          100        0       ?

Route Distinguisher: 192.168.2.4:110      (L2VNI 110)
*>i [2]:[0]:[0]:[00:50:79:66:68:07]:[]                     192.168.2.4     0          100        0       ?
* i [2]:[0]:[0]:[00:50:79:66:68:07]:[]                     192.168.2.4     0          100        0       ?
*>i [3]:[0]:[192.168.2.4]                                  192.168.2.4     0          100        0       ?
* i [3]:[0]:[192.168.2.4]                                  192.168.2.4     0          100        0       ?
Total number of entries 6
```

- Validate VXLAN traffic is seen in the wireshark capture



- Validate EVPN mac address advertisements



Appendix A – Complete Configurations

- If you face issues during your lab, you can verify your configs with the configs listed in this section
- If configs are the same, try powering off/powering on the switches to reboot them

Host1

```
VPCS> show ip

NAME        : VPCS[1]
IP/MASK     : 10.0.110.1/24
GATEWAY     : 10.0.110.254
DNS         :
MAC         : 00:50:79:66:68:05
LPORT       : 20000
RHOST:PORT  : 127.0.0.1:30000
MTU         : 1500
```

Host2

```
VPCS> show ip

NAME        : VPCS[1]
IP/MASK     : 10.0.110.2/24
GATEWAY     : 10.0.110.254
DNS         :
MAC         : 00:50:79:66:68:07
LPORT       : 20000
RHOST:PORT  : 127.0.0.1:30000
MTU         : 1500
```

Leaf1

```
Leaf1(config)# show run
Current configuration:
!
!Version ArubaOS-CX Virtual.10.07.0004
!export-password: default
hostname Leaf1
user admin group administrators password ciphertext
AQBapSPZp78qvC94j6b3r6cBrCT4vpXIZiAtwZlk+YlEmnu3YgAAAKxOjzj7QIg23YpjdsYO+48GczjpQ5zSXLeRoQQe
yWLbPhKUGKs5HoRlYiqkW1nHVH35KYbOflVG0YSjmEsE/sZ8m5fymhMh
fL/s1PPcBoahvPCPKUc0xv8Y1jpuSPg9oTZc
led locator on
no usb
ntp server pool.ntp.org minpoll 4 maxpoll 4 iburst
ntp enable
!
!
!
!
radius-server host 10.10.0.105 key ciphertext
AQBapdAz4irjSK61Zg/CFArsNYWKbn1LObqDD/v9SH1eMQ6ABQAAADY26liu vrf mgmt
!
!
!
ssh server vrf mgmt
vlan 1,110
```

```
evpn
    vlan 110
        rd auto
        route-target export auto
        route-target import auto
interface mgmt
    no shutdown
    ip static 10.10.0.122/24
    default-gateway 10.10.0.1
port-access role role1
    vlan trunk allowed 110
aaa authentication port-access mac-auth
    enable
interface 1/1/1
    no shutdown
    no routing
    vlan access 1
    aaa authentication port-access client-limit 5
    aaa authentication port-access mac-auth
        enable
interface 1/1/2
    no shutdown
    ip address 192.168.4.1/31
    ip ospf 1 area 0.0.0.0
    ip ospf network point-to-point
interface 1/1/3
    no shutdown
    ip address 192.168.4.5/31
    ip ospf 1 area 0.0.0.0
    ip ospf network point-to-point
interface 1/1/4
    no shutdown
interface 1/1/5
    no shutdown
interface 1/1/6
    no shutdown
interface loopback 0
    ip address 192.168.2.3/32
    ip ospf 1 area 0.0.0.0
interface vlan 110
interface vxlan 1
    source ip 192.168.2.3
    no shutdown
    vni 110
        vlan 110
!
!
!
!
!
router ospf 1
    router-id 192.168.2.3
    area 0.0.0.0
router bgp 65001
    bgp router-id 192.168.2.3
    neighbor 192.168.2.1 remote-as 65001
    neighbor 192.168.2.1 update-source loopback 0
    neighbor 192.168.2.2 remote-as 65001
    neighbor 192.168.2.2 update-source loopback 0
    address-family l2vpn evpn
        neighbor 192.168.2.1 activate
        neighbor 192.168.2.1 send-community extended
        neighbor 192.168.2.2 activate
        neighbor 192.168.2.2 send-community extended
    exit-address-family
```

```
   !
https-server vrf mgmt
```

Leaf2

```
Leaf2# show run
Current configuration:
!
!Version ArubaOS-CX Virtual.10.07.0004
!export-password: default
hostname Leaf2
user admin group administrators password ciphertext
AQBapfookpxvh5DM2qek95LChaPiZFKVOPVFlu6y8vK5XzeJYgAAAEz3ZqwT6rM8PgWTMwpD1zWncpp4KVh4NGlgPJxTumZ
gOItJq5pkJQxQDa63lI0Kb4qA4n5vrlkEAqJk0mDi1TWDTu2uQhnB
4gzhSOJCZ5gi07JomKdyFHHVOq0MMFiIC0Qa
led locator on
no usb
ntp server pool.ntp.org minpoll 4 maxpoll 4 iburst
ntp enable
!
!
!
!
radius-server host 10.10.0.105 key ciphertext
AQBapdAz4irjSK61Zg/CFArsNYWKbn1LObqDD/v9SH1eMQ6ABQAAADY26liu vrf mgmt
!
!
!
ssh server vrf mgmt
vlan 1,110
evpn
    vlan 110
        rd auto
        route-target export auto
        route-target import auto
interface mgmt
    no shutdown
    ip static 10.10.0.120/24
    default-gateway 10.10.0.1
port-access role role1
    vlan trunk allowed 110
aaa authentication port-access mac-auth
    enable
interface 1/1/1
    no shutdown
    no routing
    vlan access 1
    aaa authentication port-access client-limit 5
    aaa authentication port-access mac-auth
        enable
interface 1/1/2
    no shutdown
    ip address 192.168.4.3/31
    ip ospf 1 area 0.0.0.0
    ip ospf network point-to-point
interface 1/1/3
    no shutdown
```

```
    ip address 192.168.4.7/31
    ip ospf 1 area 0.0.0.0
    ip ospf network point-to-point
interface 1/1/4
    no shutdown
interface 1/1/5
    no shutdown
interface 1/1/6
    no shutdown
interface loopback 0
    ip address 192.168.2.4/32
    ip ospf 1 area 0.0.0.0
interface vlan 110
interface vxlan 1
    source ip 192.168.2.4
    no shutdown
    vni 110
        vlan 110
!
!
!
!
!
router ospf 1
    router-id 192.168.2.4
    area 0.0.0.0
router bgp 65001
    bgp router-id 192.168.2.4
    neighbor 192.168.2.1 remote-as 65001
    neighbor 192.168.2.1 update-source loopback 0
    neighbor 192.168.2.2 remote-as 65001
    neighbor 192.168.2.2 update-source loopback 0
    address-family l2vpn evpn
        neighbor 192.168.2.1 activate
        neighbor 192.168.2.1 send-community extended
        neighbor 192.168.2.2 activate
        neighbor 192.168.2.2 send-community extended
    exit-address-family
!
https-server vrf mgmt
```

Spine1

```
Spine1# show run
Current configuration:
!
!Version ArubaOS-CX Virtual.10.07.0004
!export-password: default
hostname Spine1
user admin group administrators password ciphertext
AQBapVfPYNriFKtbaTxf6gZ+VDUpQHxCp/bSZdYdYHZADp0OYgAAAB9kK0TgFlfEiblCwaaE6OllITKWVIG8Kpy8IQ8r3G4
9rijqjB7LhY8ACHJLfUj/qoXTwDAGBUjtY8n/B5lBpTdWR39u1bIuLlW
1pYCQBRucybBZvDf2Qjnbyb+tnQFokFKB
led locator on
no usb
ntp server pool.ntp.org minpoll 4 maxpoll 4 iburst
ntp enable
!
```

```
!
!
!
!
!
ssh server vrf mgmt
vlan 1
interface mgmt
    no shutdown
    ip dhcp
interface 1/1/1
    no shutdown
    ip address 192.168.4.2/31
    ip ospf 1 area 0.0.0.0
    ip ospf network point-to-point
interface 1/1/2
    no shutdown
    ip address 192.168.4.0/31
    ip ospf 1 area 0.0.0.0
    ip ospf network point-to-point
interface 1/1/3
    no shutdown
interface 1/1/4
    no shutdown
interface 1/1/5
    no shutdown
interface 1/1/6
    no shutdown
interface loopback 0
    ip address 192.168.2.1/32
    ip ospf 1 area 0.0.0.0
!
!
!
!
!
router ospf 1
    router-id 192.168.2.1
    area 0.0.0.0
router bgp 65001
    bgp router-id 192.168.2.1
    neighbor 192.168.2.3 remote-as 65001
    neighbor 192.168.2.3 update-source loopback 0
    neighbor 192.168.2.4 remote-as 65001
    neighbor 192.168.2.4 update-source loopback 0
    address-family l2vpn evpn
        neighbor 192.168.2.3 activate
        neighbor 192.168.2.3 route-reflector-client
        neighbor 192.168.2.3 send-community extended
        neighbor 192.168.2.4 activate
        neighbor 192.168.2.4 route-reflector-client
        neighbor 192.168.2.4 send-community extended
    exit-address-family
!
https-server vrf mgmt
```

Spine2

```
Spine2# show run
Current configuration:
!
!Version ArubaOS-CX Virtual.10.07.0004
!export-password: default
hostname Spine2
user admin group administrators password ciphertext AQBapRCkK24BKPCzJkYIe2XC31Moe9Pb5j
Y8DJW/vigce5rLYgAAAO0fR1zTF05CEXxGkkwGkXODA9JOu4S26OMJvzRi3DS+v/H7lLhaPlO66OTQHaYhvVuX
5QrZtwk5jvMmgzasPg6sSO48r8o9ajdVz30tgwuSYUXuOPKVaY8JFTLKyJewDqnb
led locator on
no usb
ntp server pool.ntp.org minpoll 4 maxpoll 4 iburst
ntp enable
!
!
!
!
!
!
ssh server vrf mgmt
vlan 1
interface mgmt
    no shutdown
    ip dhcp
interface 1/1/1
    no shutdown
    ip address 192.168.4.6/31
    ip ospf 1 area 0.0.0.0
    ip ospf network point-to-point
interface 1/1/2
    no shutdown
    ip address 192.168.4.4/31
    ip ospf 1 area 0.0.0.0
    ip ospf network point-to-point
interface 1/1/3
    no shutdown
interface 1/1/4
    no shutdown
interface 1/1/5
    no shutdown
interface 1/1/6
    no shutdown
interface loopback 0
    ip address 192.168.2.2/32
    ip ospf 1 area 0.0.0.0
!
!
!
!
!
router ospf 1
    router-id 192.168.2.2
    area 0.0.0.0
router bgp 65001
    bgp router-id 192.168.2.2
```

```
    neighbor 192.168.2.3 remote-as 65001
    neighbor 192.168.2.3 update-source loopback 0
    neighbor 192.168.2.4 remote-as 65001
    neighbor 192.168.2.4 update-source loopback 0
    address-family l2vpn evpn
        neighbor 192.168.2.3 activate
        neighbor 192.168.2.3 route-reflector-client
        neighbor 192.168.2.3 send-community extended
        neighbor 192.168.2.4 activate
        neighbor 192.168.2.4 route-reflector-client
        neighbor 192.168.2.4 send-community extended
    exit-address-family
!
https-server vrf mgmt
```

Appendix B – ClearPass Installation in EVE-NG

Pre-Requisites:

- An Aruba Support Port account will be required to download the ClearPass OVA as well as EVAL licenses.

Steps

1. To first install the ClearPass OVA into the EVE-NG environment, follow the instructions at this link:

   https://www.eve-ng.net/index.php/documentation/howtos/howto-add-aruba-clearpass/

   This lab uses the latest ClearPass OVA v. 6.9.0, which can be downloaded from the Aruba Support Portal:

   https://asp.arubanetworks.com/downloads

2. Once installed, and the node is created in the EVE-NG lab file, follow the configuration steps for ClearPass. First login to ClearPass using the default credentials (appadmin/eTIPS123). Once entered, the configuration process will begin.

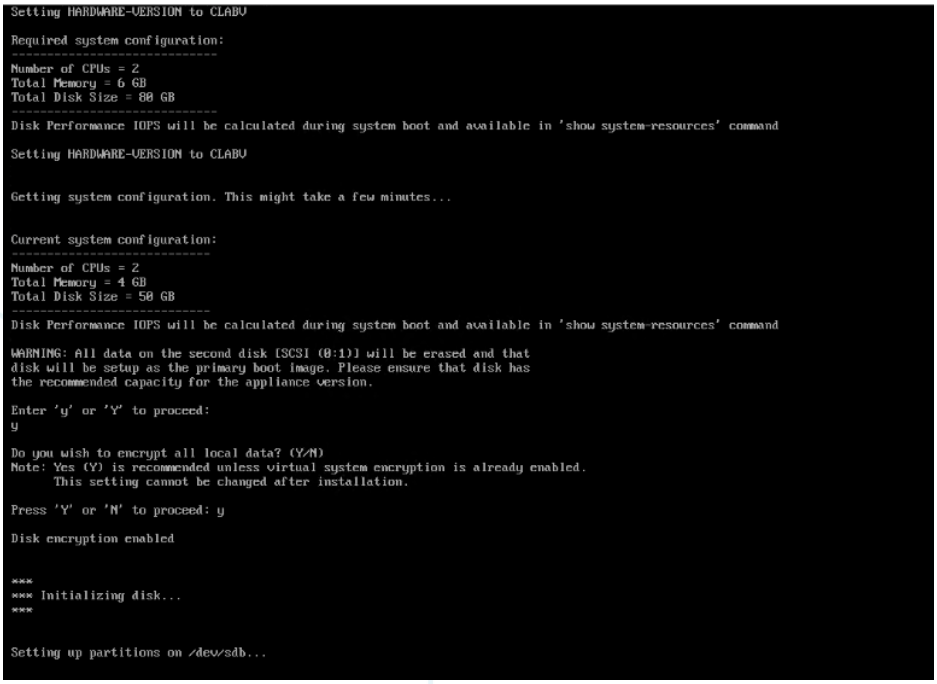


Figure 19. ClearPass Installation

Select the CLABV installation, click “Y” to proceed and “Y” to encrypt data.

3. Once prompted, enter the IP address as “10.10.0.105”, the mask as “255.255.255.0”, the gateway as “10.10.0.254”, and the DNS as “8.8.8.8” (not needed for this exercise). Configure a new password, this lab example used “aruba123”.

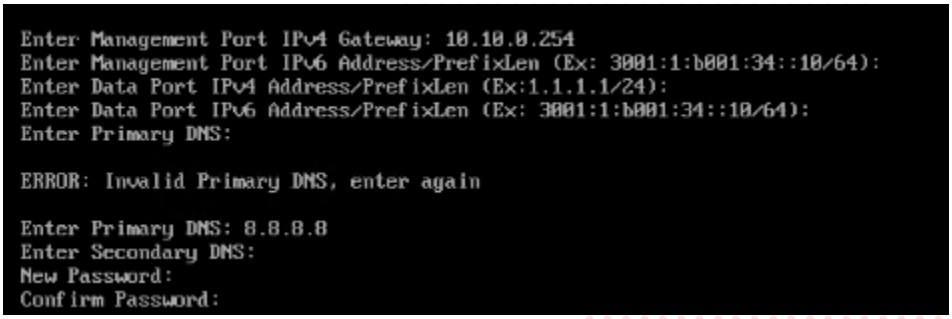


Figure 20. ClearPass IP Configuration

4. Configure the date and time manually as well as the time zone.

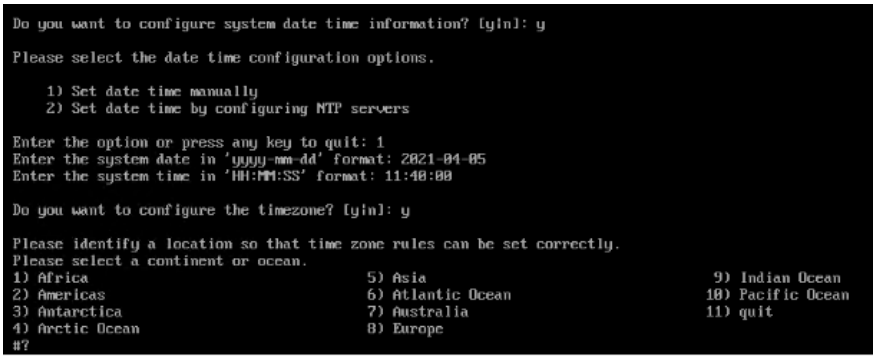


Figure 21. ClearPass Date and Time Configuration

5. Confirm the correct date, time, and time zone.

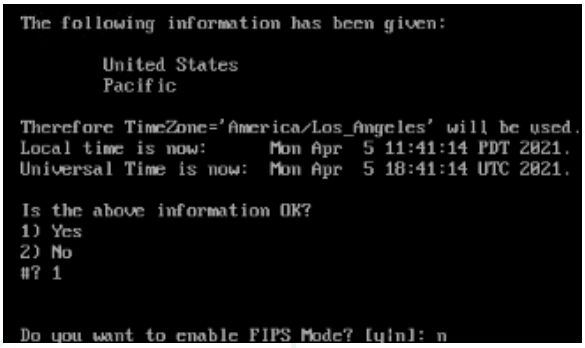


Figure 22. ClearPass Date and Time Settings Confirmation

6. Confirm the configured settings are correct. Press Y to save settings.

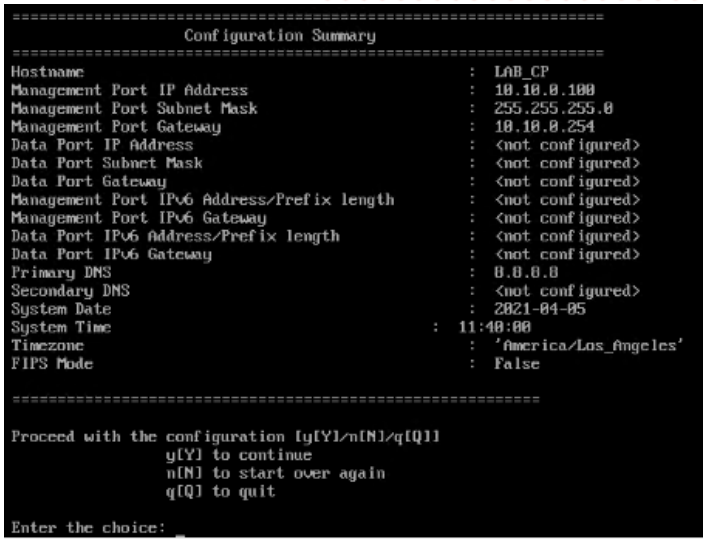


Figure 23. ClearPass Configuration Confirmation

7. ClearPass will then reboot and will then allow the user to log in to add licenses. Enter the platform license key retrieved from the Aruba Support Portal Licensing Management System - https://lms.arubanetworks.com/.

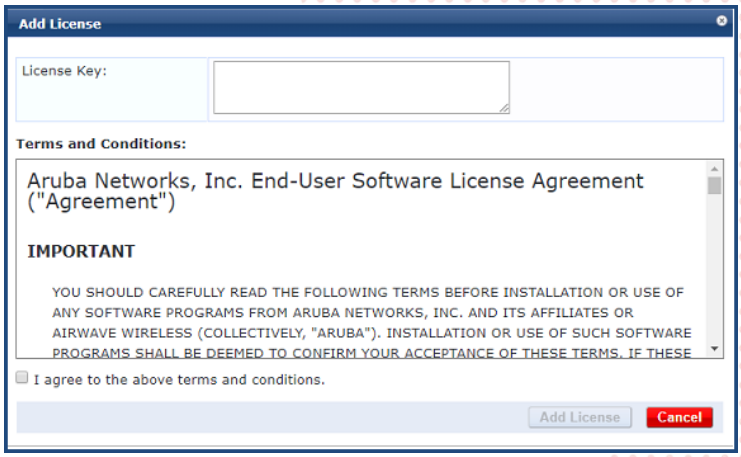


Figure 24. ClearPass Platform License entry

8. Once logged into ClearPass, enter the licensing section (Administration → Server Manager → Licensing). Click on "Add License".

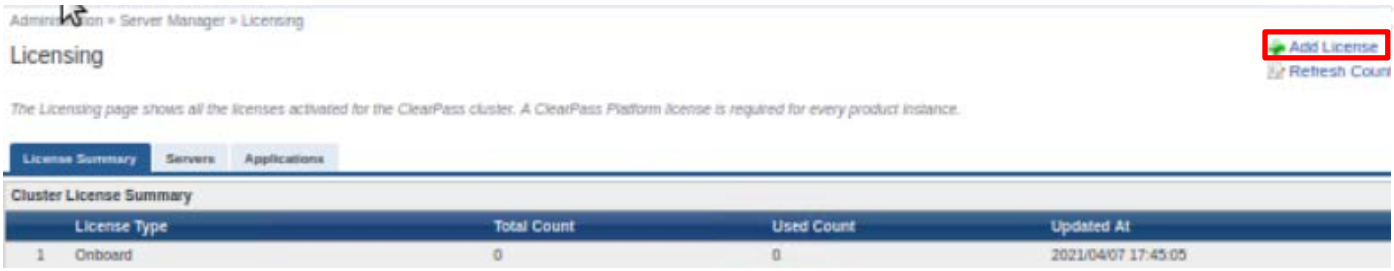


Figure 25. ClearPass Add New Server License

9. Add the new license and agree to the terms and conditions. ClearPass will then be ready to configure for authentication.

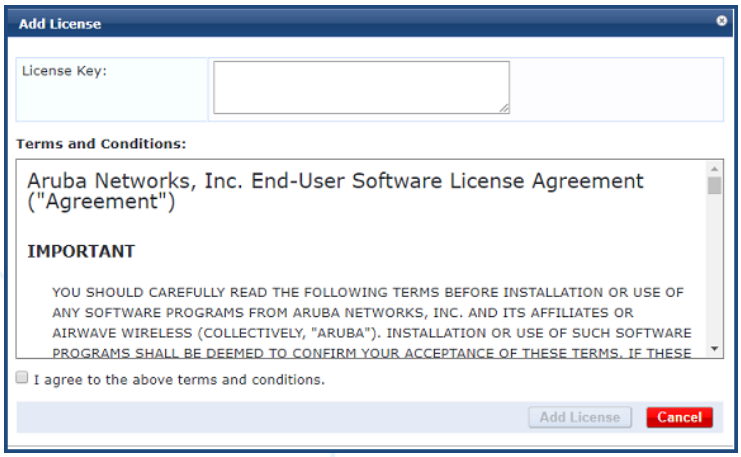


Figure 26. ClearPass Server license entry

a Hewlett Packard Enterprise company

www.arubanetworks.com

3333 Scott Blvd. Santa Clara, CA 95054
1.844.472.2782 | T: 1.408.227.4500 | FAX: 1.408.227.4550 | info@arubanetworks.com