SDWOTS

SD-WAN Operations and Troubleshooting

Version 19.2

Date: June 26, 2020

Table of Contents

Lab 0: Access the Lab Devices **3**

- Task 1: Access the NterOne Lab 3

Lab 1: Reset vEdge Router **6**

Lab 2: Remove vEdge from vManage Inventory **17**

Lab 3: Add vEdge to vManage Inventory **24**

Lab 4: Configure and Deploy Control-Plane **29**

Lab 5: Configure and Deploy an Overlay Network **44**

Lab 6: Exploring vManage GUI **56**

Lab 7: Provision and Deploy vManage Templates **104**

- Task 1: Deploy Feature-Based Device Template 109
- Task 2: Deploy CLI Based Device Template 143

Lab 8: Provision and Deploy Policies (Part 1) **156**

- Task 1: Modify Controller Distribution using a Control Policy 157
- Task 2: Create a Hub and Spoke Control Policy 175

Lab 9: Provision and Deploy Policies (Part 2) **179**

- Task 1: Create an Application Aware Routing Policy 180
- Task 2: Create a Service Chain Policy 207
- Task 3: Create Localized QOS Policies Using Feature Templates 224

Lab 10: Verifying and Troubleshooting Data Plane Issues **230**

- Task 1: Verifying the Data Plane 231
- Task 2: Verifying BFD Information 232
- Task 3: Determining the Makeup of Network Traffic Using DPI 234
- Task 4: Simulating Flows 236

Lab 11: TLOC Extension **239**

Lab 12: Troubleshooting Methods and Tools **257**

- Task 1: Ping and Traceroute 258
- Task 2: vManage Alarms and Events 261
- Task 3: vShell 262
- Task 4: Viewing System Log Files 263
- Task 5: Enabling and View Debug Output 265

Lab 13: Deploy vEdge Router using ZTP (Zero Touch Provisioning) & (Troubleshooting) 267

- Task 1: Deploy ZTP Server 268
- Task 2: Deploy vEdge Router Using ZTP 277

Lab 14: Using the REST API and Python Scripting 303

- Task 1: Python Installation and Introduction 303
- Task 2: Examine and Run a Sample Python Script and View the Output 310

Lab 15: Deploy Multitenant vManage 314

- Task 1: Deploy Multitenant vManage 315
- Task 2: Install vManage Certificates 324
- Task 3: Add vBond Controller and Install Certificates 330
- Task 4: Add a Tenant in vManage 338
- Task 5: Access vManage as a Tenant 351

Appendix A 355

- Lab A1 - Decommission a cEdge Router from vManage Inventory 355
- Lab A2 – Adding a cEdge Router to vManage Inventory 360

 © 2020 NterOne Corporation

Lab 0: Access the Lab Devices

In this this activity, you will access the NterOne Lab and login to your assigned pod using credentials supplied by your instructor.

Activity Objectives

In this activity, you will access the NterOne Lab environment. The following values will be used in this Lab.

Task 1: Access the NterOne Lab

In this task, you will access the NterOne Lab from your computer using the credentials supplied by your instructor. Enter the credentials and URL in the tables below:

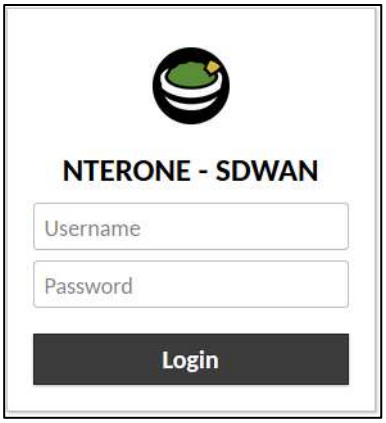
Username:	Password:

Access URL:
https://labs.nocular.com/#/

Activity Procedure

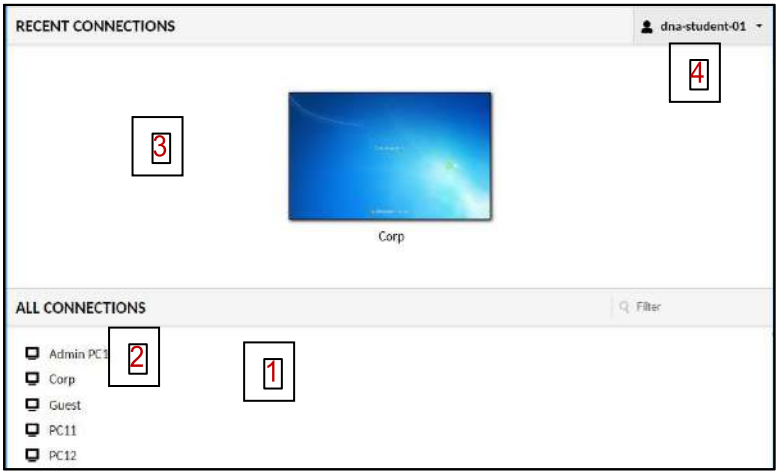
Complete these steps:

Step 1 Use a web browser of your choice, and browse to the **Lab URL** provided by your instructor. You should see the login screen below:



Step 2 Enter the credentials provided by your instructor and click **Login.**

Step 3 Observe the list of connections available **(1),** the type of devices **(2)**, the Recent Connections pane **(3),** and the Menu Bar **(4).**



Step 4 Each device in your pod is listed, alphabetically, under "ALL CONNECTIONS" in the lower half of the screen

Step 5 A connection with a monitor icon provides a desktop, either through RDP or VNC; a connection with the prompt icon (>_) provides a **Telnet or SSH console**

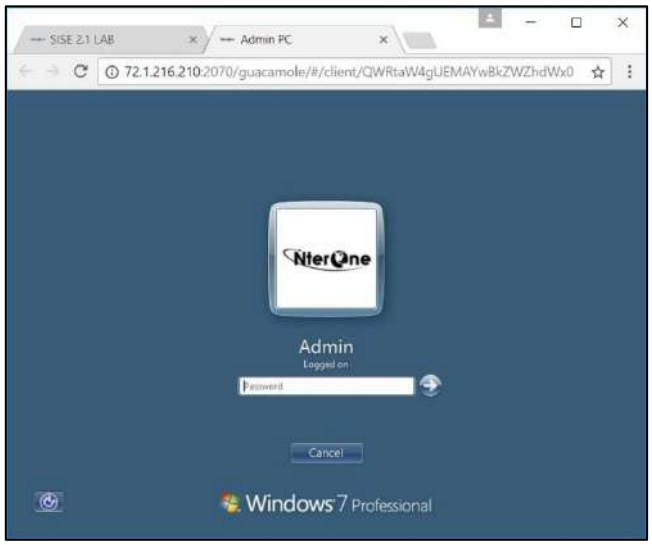
Step 6 For your convenience, recently accessed devices appear under "RECENT CONNECTIONS" at the top of the screen

Step 7 The menu in the upper right shows the username you are logged in under. Click the downwards arrow to open a dropdown menu, to access settings and to logout.

Step 8 Try opening a session**. Right-click** on a server or PC (monitor icon), and select **open link in new tab**. Alternatively, hold down **COMMAND/CONTROL** and **left-click** the device to open the connection in a new tab.

 © 2020 NterOne Corporation

Step 9 The connection opens in a new tab. Login to the device. The Admin-PC's Admin account username is **admin** and password is **admin.**



Step 10 **Use the onboard keyboard to provide access to the CTRL-ALT-DEL key sequence. To access the on-board keyboard press <u>CTRL-ALT-SHIFT</u>**

Activity Verification

You have completed this task when you obtain the following results:

- Successfully logged into the NterOne Lab environment
- Accessed a device via RDP or VNC or SSH

Lab 1: Reset vEdge Router

In this Lab, you will use CLI commands and the vManage GUI to reset your vEdge Cloud Router.

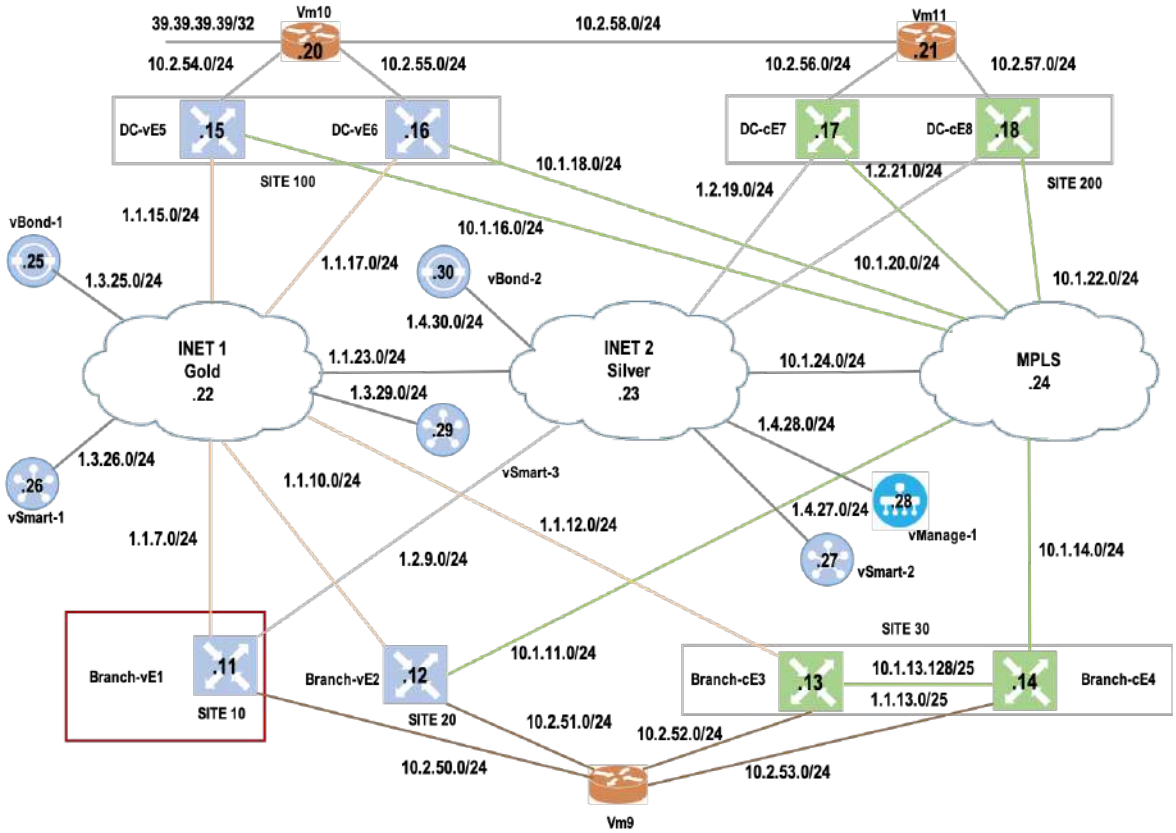
Activity Objectives

In this activity, you will reset your pod's vEdge Cloud router.

Device Name	IP Address or URL	Username	Password	LAB
Branch-vE1	1.1.7.11 – SSH 10.82.83.111	admin	admin	NterOne Lab
vManage-1	https://1.4.28.28 – SSH 10.82.83.131	admin	admin	NterOne Lab

 © 2020 NterOne Corporation

Topology



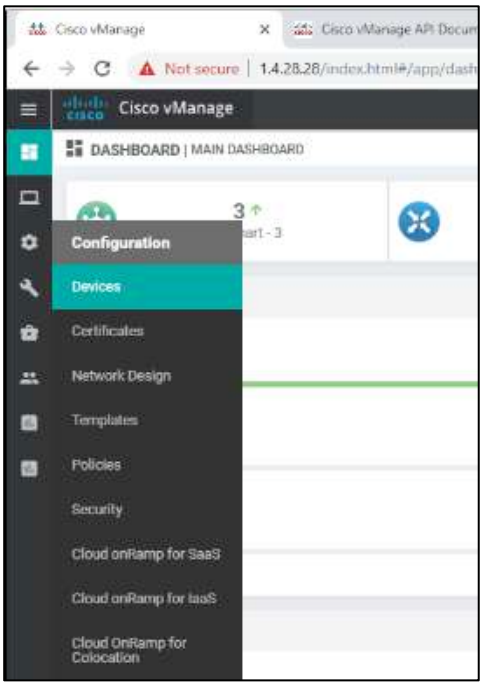
Activity Procedure

Complete these steps:

Step 1 **First, use SSH to connect to your Branch-vE1 Cloud Router**. *(Refer to the table at the beginning of the lab)*

Step 2 **Second, open the Chrome browser window and login to vManage** (https://1.4.28.28:8443 (U:admin, P:admin).

Step 3 Go to **Configuration > Devices** and from the list find and highlight the vEdge router **branch-vE1.** Make sure to select the correct device.



Step 4 Identify the correct vEdge router from the Device list.

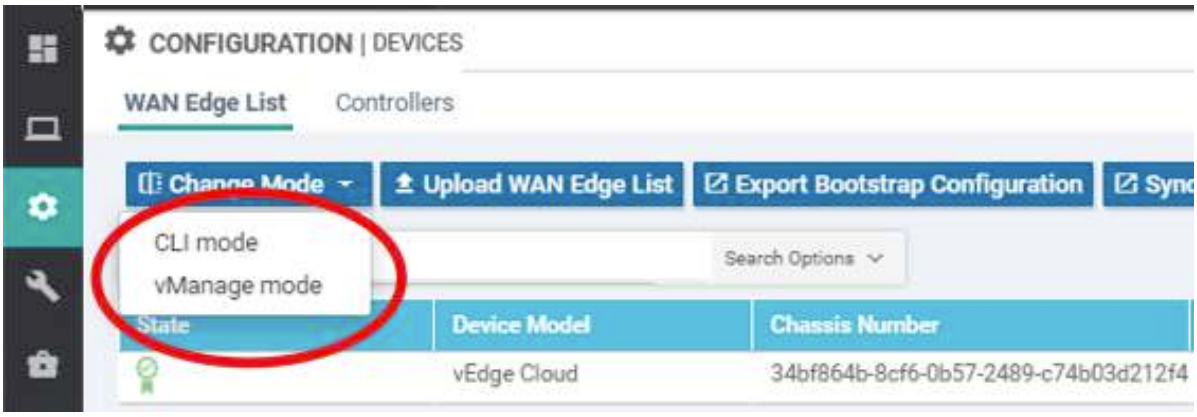
Step 5 If the Mode is vManage then it needs to be changed to CLI. <u>If it's already CLI skip to Step 9.</u>

Note When a device is in vManage mode, it means that it is attached to a Device Template.

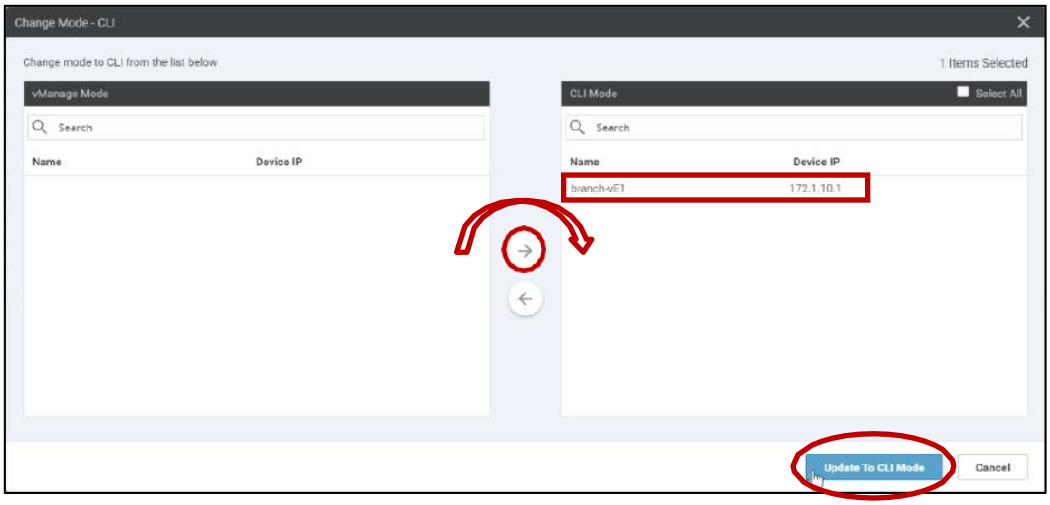
 © 2020 NterOne Corporation

Hostname	System IP	Site ID	Mode
branch-vE1	172.1.10.1	10	vManage

Step 6 Make sure that **branch-vE1 is still selected**, then go to the top left corner of the window and click **Change Mode**, and then select **CLI mode**.



Step 7 The **Change Mode - CLI** window will appear. Select the **CORRECT** vEdge device from the **vManage Mode** list and add it to the right under **CLI Mode**. Click **Update To CLI Mode** when done.



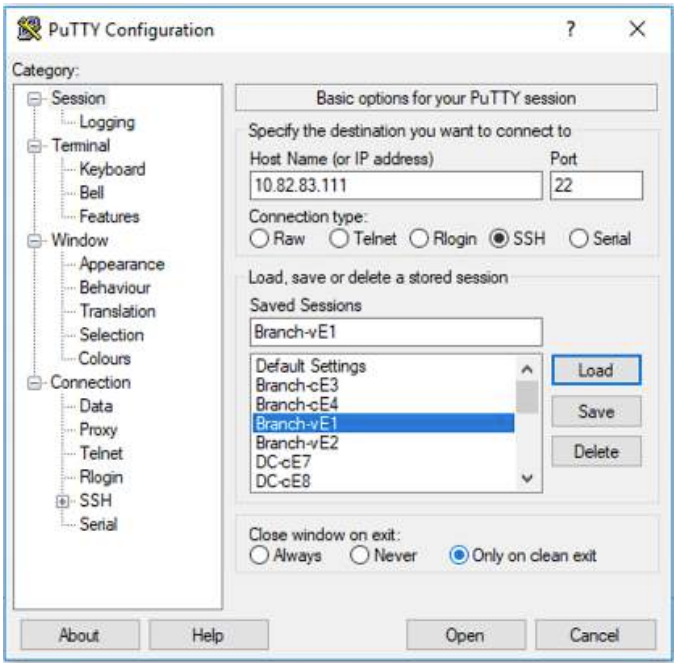
Step 8 Wait patiently periodically pressing the refresh button

Step 9 Go back to **Configuration → Device and confirm on the Devices list that it has been changed to CLI mode**. Contact your instructor if you encounter any problems.

branch-vE1	172.1.10.1	10	CLI

Step 10 Open Putty and enter **10.82.83.111** (the address of the **Branch-vE1** router) or select **branch-vE1** from the saved sessions. Press the **Open** button to

open the SSH session. Login with a username of **admin** and a password of **admin**



Step 11 Navigate to SDWAN Lab Files folder on the Admin PC desktop, and open the "**Branch-vE1-RESET.txt"** file to copy and paste all of the text into your CLI. You may need to add a line "yes" after commit for the script to run correctly. The router should reboot on its own.

 © 2020 NterOne Corporation

```
branch-vE1# ***Paste*** (Right-Click in Putty)
```

Step 12 **DO NOT REMOVE VPN 512** otherwise this will disconnect your SSH session and you will not be able to continue with this Lab. VPN 512 is the management interface, if this were deleted, the instructor would have to go the console in ESXi to repair VPN 512 connectivity.

Step 13 If the script did not reboot the vEdge router, then make sure to reboot it and verify the configuration. After the reboot, the router prompt should be - **vedge#**.

Step 14 vEdge# **reboot** (If not already reloaded) You will lose ssh connectivity.

Step 15 Reconnect to the CLI of the branch-vE1 router and issue the command **show run** to verify that the configuration has been reset.

```
vedge# show run
 system
  host-name                vedge
  admin-tech-on-failure
  route-consistency-check
  aaa
   auth-order local radius tacacs
   usergroup basic
    task system read write
    task interface read write
   !
   usergroup netadmin
   !
   usergroup operator
    task system read
    task interface read
    task policy read
    task routing read
    task security read
   !
   usergroup tenantadmin
   !
   user admin
    password
$6$lV1XHpFjQrUBVnS1$5zj18TA8WR5oxNGGoScmpUtrL81xFh61qvytKxdkQf
rO9n.72K/j2IiObyz4hLajklMqBe84DRHYQNyXexEta0
   !
  !
  logging
   disk
    enable
   !
  !
 !
 omp
  shutdown
  graceful-restart
 !
 vpn 0
 !
 vpn 512
  interface eth0
   ip address 10.82.83.111/20
   no shutdown
!
```

Step 16 An SD-WAN deployment assumes a **Zero Trust model** is in place prior to any control-plane connectivity.

Use the command **show control local-properties** to verify System/Control settings.

The **Certificate** has not been installed. A trusted certificate is required in order to establish a connection to the **vBond** orchestrator, which will provide the necessary information to establish a connection with the **vSmart** controllers and **vManage NMS**.

Also, note that there is no **serial-num** associated to this vEdge router. The **serial-num** will be filled in once a certificate is installed.

For **vEdge Cloud** a one-time password in the form of a **token** is required to establish an initial connection with the **vBond** orchestrator.

```
vedge# show control local-properties
 personality                        vedge
 sp-organization-name
 organization-name
 root-ca-chain-status               Installed

 certificate-status                 Not-Installed
 certificate-validity               Not Applicable
 certificate-not-valid-before       Not Applicable
 certificate-not-valid-after        Not Applicable

 dns-name
 site-id                            0
 domain-id                          1
 protocol                           dtls
 tls-port                           0
 system-ip                          0.0.0.0
 chassis-num/unique-id              34bf864b-8cf6-0b57-2489-
c74b03d212f4
 serial-num                         No certificate installed
 token                              Invalid
 keygen-interval                    1:00:00:00
 retry-interval                     0:00:00:19
 no-activity-exp-interval           0:00:00:20
 dns-cache-ttl                      0:00:02:00
 port-hopped                        FALSE
 time-since-last-port-hop           0:00:00:00
```

 © 2020 NterOne Corporation

```
embargo-check                    success
number-vbond-peers               0
number-active-wan-interfaces     0
```

Step 17 Notice the **chassis-num/unique-id**. You will use this value in later steps to identify the correct vEdge router and to confirm **vEdge Cloud Router Activation**. The following value should be used but verify with your Instructor if there are any changes to the LAB.

34bf864b-8cf6-0b57-2489-c74b03d212f4 (Chassis number)

Step 18 Although not recommended, certificates can be installed manually if necessary.

Later in the lab you will use **vManage** to generate a **Token,** and along with the **Chassis Number** they will be used to establish a **secure connection** with the vBond orchestrator.

Once a secure connection is established with vBond, a certificate will be automatically installed on the **vEdge Cloud** router (Default vManage behavior).

Step 19 Make the Putty session as wide as the screen will allow and further verify the configuration, issue the following commands.

```
vEdge# show omp peers

vEdge# show control connections

vEdge# show certificate installed

vedge# show interface | tab

                                    IF      IF      IF
                AF                  ADMIN   OPER    TRACKER  ENCAP  PORT
 VPN  INTERFACE  TYPE  IP ADDRESS   STATUS  STATUS  STATUS   TYPE   TYPE     MTU
 ---------------------------------------------------------------------------------
-
 0    ge0/0      ipv4  -            Down    Down    NA       null   service  1500
 0    ge0/1      ipv4  -            Down    Down    NA       null   service  1500
 0    ge0/2      ipv4  -            Down    Down    NA       null   service  1500
 0    ge0/3      ipv4  -            Down    Down    NA       null   service  1500
 512  eth0       ipv4  10.82.83.111/20 Up   Up      NA       null   service  1500

vedge# show ip route
```

```
Codes Proto-sub-type:
  IA -> ospf-intra-area, IE -> ospf-inter-area,
  E1 -> ospf-external1, E2 -> ospf-external2,
  N1 -> ospf-nssa-external1, N2 -> ospf-nssa-external2,
  e -> bgp-external, i -> bgp-internal
Codes Status flags:
  F -> fib, S -> selected, I -> inactive,
  B -> blackhole, R -> recursive

                                      PROTOCOL  NEXTHOP   NEXTHOP       NEXTHOP
VPN    PREFIX            PROTOCOL     SUB TYPE  IF NAME   ADDR          VPN
-----------------------------------------------------------------------------------
512    10.82.80.0/20     connected     -        eth0       -            -        -
```

Step 20 Use the chart below and Configure the **System** parameters. These values **MUST** be entered correctly. **(Case-sensitive)**

Command	Value
host-name	branch-vE1
system-ip	172.1.10.1
site-id	10
controller-group-list	10
organization-name	“NterOne Lab”
vbond	vbond.demo.local
upgrade-confirm	5

```
vedge# config
 Entering configuration mode terminal
 vedge(config)# system
 vedge(config-system)# host-name branch-vE1
 vedge(config-system)# system-ip 172.1.10.1
 vedge(config-system)# site-id 10
 vedge(config-system)# controller-group-list 10
 vedge(config-system)# organization-name "NterOne Lab"
 vedge(config-system)# vbond vbond.demo.local
 vedge(config-system)# upgrade-confirm 5
 vedge(config-system)# show config
```

 © 2020 NterOne Corporation

```
 system
  host-name                 branch-vE1
  system-ip                 172.1.10.1
  site-id                   10
  controller-group-list     10
  organization-name         "NterOne Lab"
  upgrade-confirm           5
  vbond vbond.demo.local

vedge(config-system)# commit
 Commit complete.
 branch-vE1(config-system)# end
```

Step 21 Verify the system settings. Correct any value that does not match the table.

```
branch-vE1# show running-config system
 system
  host-name                 branch-vE1
  system-ip                 172.1.10.1
  site-id                   10
  controller-group-list     10
  admin-tech-on-failure
  route-consistency-check
  organization-name         "NterOne Lab"
  upgrade-confirm           5
  vbond vbond.demo.local
  aaa
   auth-order local radius tacacs
   usergroup basic
    task system read write
    task interface read write
   !
   usergroup netadmin
   !
   usergroup operator
    task system read
    task interface read
    task policy read
    task routing read
    task security read
```

```
!
usergroup tenantadmin
!
user admin
 password
$6$lV1XHpFjQrUBVnS1$5zj18TA8WR5oxNGGoScmpUtrL81xFh61qvytKxdkQfrO9n.72
K/j2IiObyz4hLajklMqBe84DRHYQNyXexEta0
!
!
logging
 disk
  enable
 !
!
!
```

Step 22 Use the **show interface** command to display the assigned interfaces. Interfaces will default to vpn 0 if not assigned to any other vpn. Take note of the system **loopback** that was created when issuing the **system-ip** command.

```
branch-vE1# show interface | tab

                                        IF      IF      IF
                AF                      ADMIN   OPER    TRACKER  ENCAP  PORT
VPN  INTERFACE  TYPE  IP ADDRESS        STATUS  STATUS  STATUS   TYPE   TYPE      MTU
-----------------------------------------------------------------------------------------
0    ge0/0      ipv4  -                Down    Down    NA       null   service   1500
0    ge0/1      ipv4  -                Down    Down    NA       null   service   1500
0    ge0/2      ipv4  -                Down    Down    NA       null   service   1500
0    ge0/3      ipv4  -                Down    Down    NA       null   service   1500
0    system     ipv4  172.1.10.1/32    Up      Up      NA       null   loopback  1500
512  eth0       ipv4  10.82.83.111/20  Up      Up      NA       null   service   1500
```

Activity Verification

You have completed this task when you attain these results:

- You have successfully reset/cleared a vEdge Cloud router.

 © 2020 NterOne Corporation

Lab 2: Remove vEdge from vManage Inventory

In this Lab, you will remove a vEdge device from the vManage inventory.

Activity Procedure

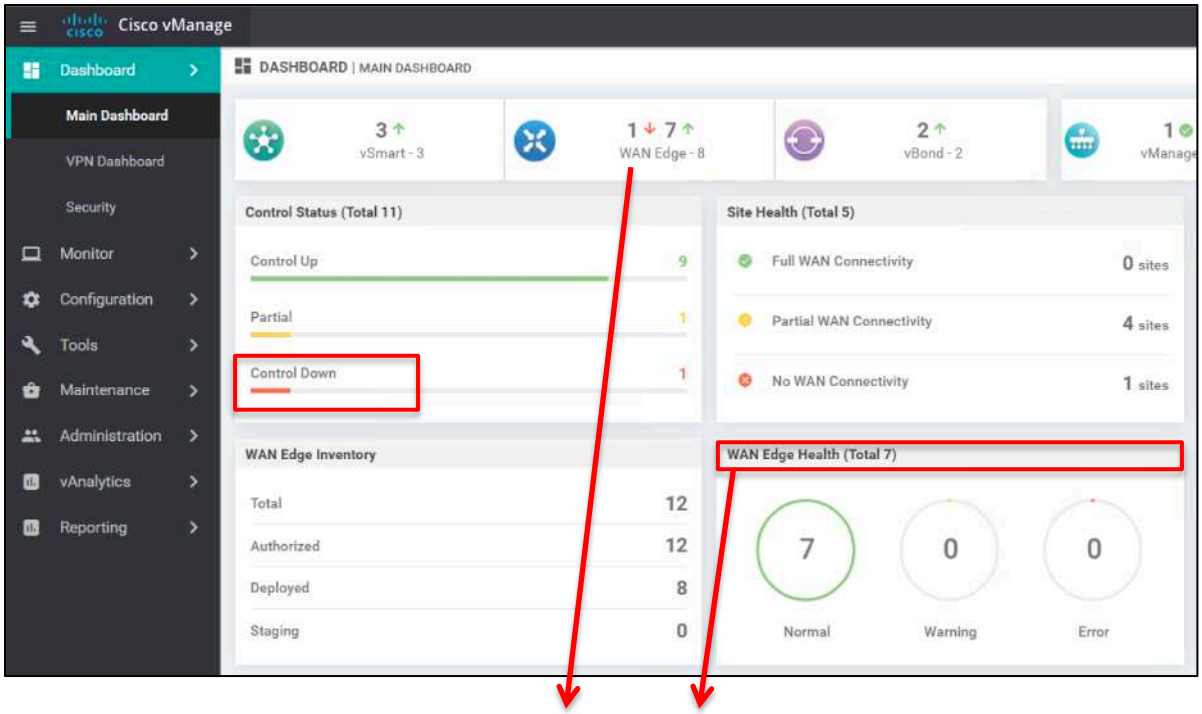
Complete these steps:

Step 1 The following vManage device will be used for this Lab.

Device Name	IP Address or URL	Username	Password	LAB
vManage-1	https://1.4.28.28 – SSH 10.82.83.131	admin	admin	NterOne Lab

Step 2 Open a browser window and log in to vManage.

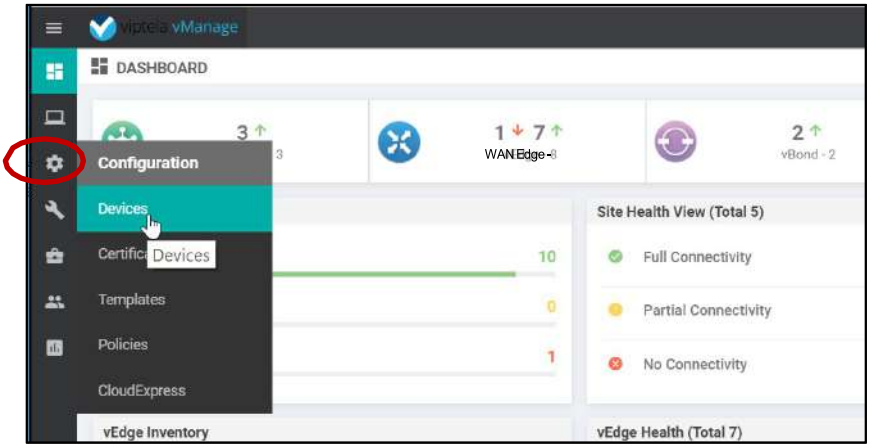
Step 3 Once logged in you will be redirected to the Dashboard. Notice, **vManage has lost connectivity with the vEdge router** that was reset in a previous Task.

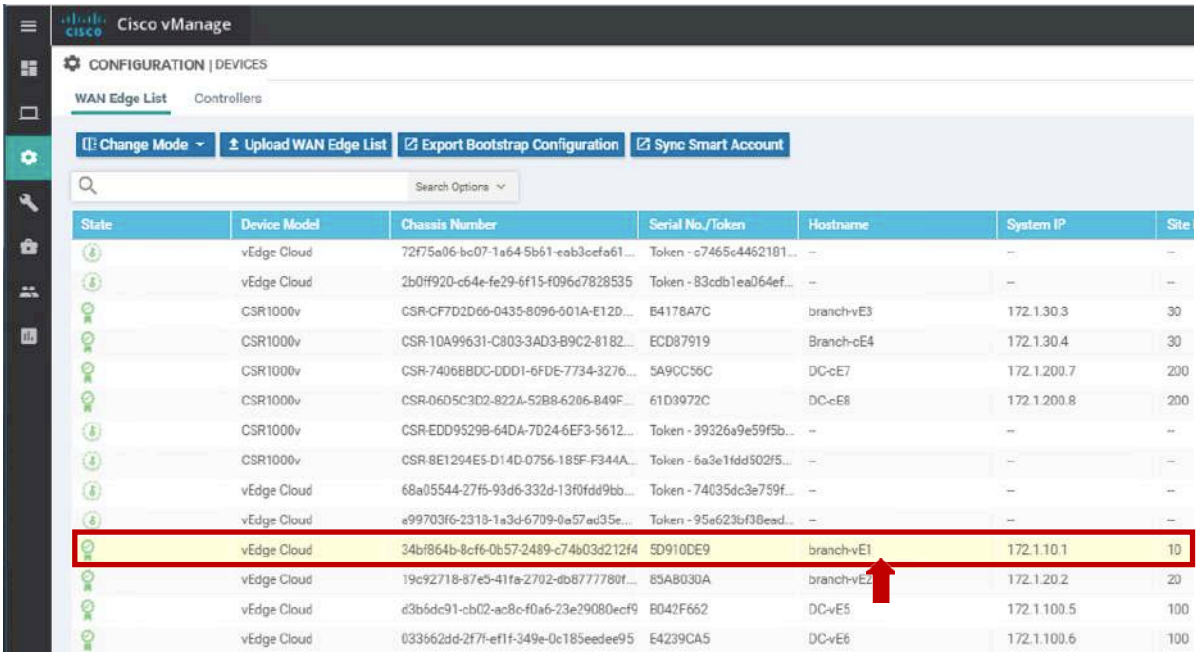


Step 4 During the next steps, you will remove the vEdge router from the vManage inventory, and then you will add the device to the inventory by uploading the **Device Serial List** using the vManage GUI. Once the list has been uploaded, vManage will generate a one-time password **(Token)** for the newly added device.

Step 5 Using the Dashboard menu, navigate to **Configuration > Devices.**

 © 2020 NterOne Corporation

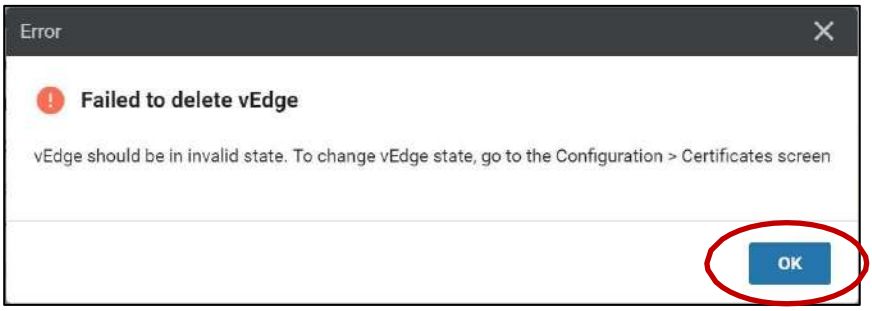




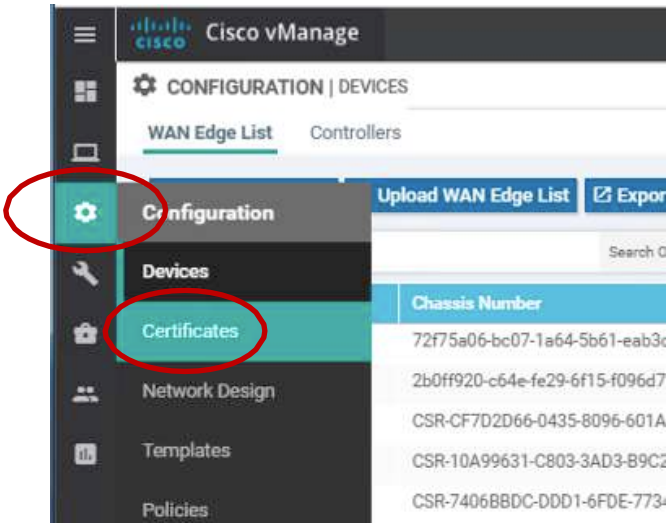
Step 6 Locate the **branch-vE1** vedge device on the list. Click on the **…** to the right of the vEdge entry and a drop down menu will appear. Select **Delete vEdge,** and then click **OK. (Please verify that you are deleting the correct router)**

Step 7 You should receive an **Error** indicating that the device cannot be deleted until it has been marked as

invalid first. Click **OK** to confirm.



Step 8 Navigate to Configuration > Certificates.



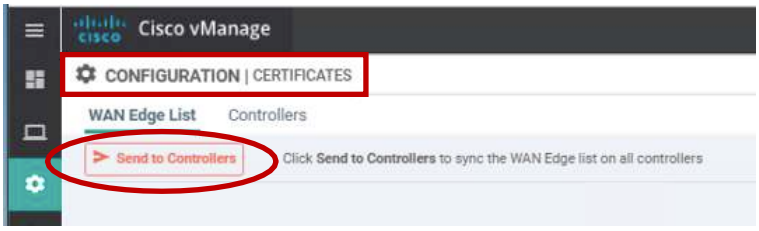
Step 9 Correctly identify the **branch-vE1** device and select the **Invalid** option all the way to the right of the vEdge entry. Click **OK** to confirm device invalidation.

 © 2020 NterOne Corporation

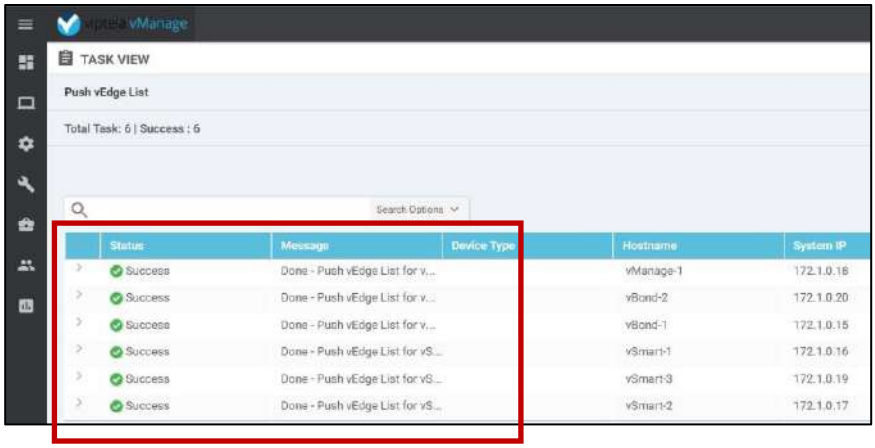
Hostname	IP Address	Validate
–	–	Invalid | Staging | Valid
–	–	Invalid | Staging | Valid
–	–	Invalid | Staging | Valid
–	–	Invalid | Staging | Valid
branch-vE1	172.1.10.1	Invalid | Staging | Valid
branch-vE2	172.1.20.2	Invalid | Staging | Valid
DC-vE5	172.1.100.5	Invalid | Staging | Valid
DC-vE6	172.1.100.6	Invalid | Staging | Valid

Step 10 Once the device has been made invalid you need to synchronize the controllers by selecting the **Send to Controllers** button at the top left of the Certificates configuration window.

Marking a device **"Invalid"** will remove it from the **valid vEdge list**, and the device will not be able to join the overlay.

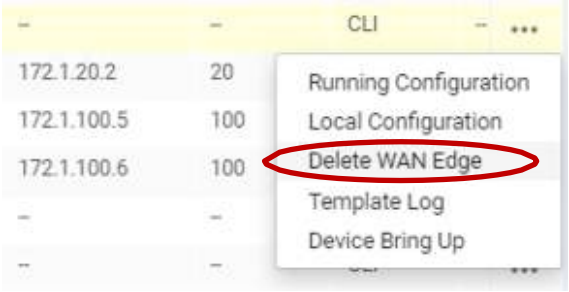


Step 11 You will be redirected to the **TASK VIEW** window where you can verify successful synchronization to the controllers. Contact your instructor if any of the tasks fail.



Step 12 Navigate to **Configuration > Certificates** to confirm that the device is **Invalid**. Make sure that the **Device Chassis Number** matches what you wrote down from a previous step.

Step 13 Navigate back to **Configuration > Devices** and **CORRECTLY** identify the vEdge device from the list (the Chassis Number ends with **d212f4**). Once again click the **…** all the way to the right of the entry and select **Delete vEdge.**



Step 14 Click **OK** to confirm vEdge deletion. Verify that the vEdge device has been purged from the inventory. When a device is deleted from the inventory it is permanently removed. In order to add the device back into the inventory you must upload the Device Serial list.

 © 2020 NterOne Corporation

State	Device Model	Chassis Number	Serial No./Token	Enterprise Cert Serial No	Enterprise Cert Expiration Date	Hostname*	System IP	Site ID	Mode	A	
	CSR1000v	CSR-10A99631-C8E9-3AD3-BFC2-81E226D39CD3	EC067919	NA	NA	Branch-vE4	172.1.30.4	30	vManage	B	...
	vEdge Cloud	19c92718-87e5-4 14a-27f02-8c877778f0a3	85AB838A	NA	NA	branch-vE2	172.1.20.2	20	vManage	B	...
	CSR1000v	CSR-CF7D2066-9435-8096-501A-E12D90AF5DAD	84178A7C	NA	NA	branch-vE3	172.1.30.3	30	vManage	B	...
	CSR1000v	CSR-740688D0-0001-6FDE-7734-327EA33086FC	549CC36C	NA	NA	DC-cE7	172.1.200.7	200	vManage	D	...
	CSR1000v	CSR-06D5C3E2-422A-6288-6206-846F7F10EB07	61D3972C	NA	NA	DC-cE8	172.1.200.8	200	vManage	D	...
	vEdge Cloud	a9e6dc91-cbf2-ed8c-f0a6-35e29d86ecf9	8042F662	NA	NA	DC-vE5	172.1.100.5	100	vManage	D	...
	vEdge Cloud	6330621d-27f7-ef1f-34fe-0c1f35ee0ee15	542390A5	NA	NA	DC-vE6	172.1.100.6	100	vManage	D	...
	vEdge Cloud	72f75a06-bc07-1a64-5b61-aab3ce6a6190	Token - c7465c44f2181417fb28af414ed22c29	NA	NA	–	–	–	CLI	–	...
	vEdge Cloud	3b0ff929-c64e-fe29-bf15-f096d7828535	Token - 60cdb1ea064ef803488b5c66e68146f7	NA	NA	–	–	–	CLI	–	...
	CSR1000v	CSR-ED093298-64DA-7024-6EF3-36128CEF968C	Token - 35926e9e59f3ad332dc43a2b532a2ed5	NA	NA	–	–	–	CLI	–	...
	CSR1000v	CSR-BE1294E5-014D-0756-185F-F344A5D6201F	Token - 6e3e1fdd502f36ede73a81bf25fc9759	NA	NA	–	–	–	CLI	–	...
	vEdge Cloud	68a05544-27f6-03d6-332d-13f0bb0bbb5	Token - 74035dc3e759ff89bff85927e4f277fc	NA	NA	–	–	–	CLI	–	...
	vEdge Cloud	a9f75bf9-2219-1a3d-5709-3a57a635ed02	Token - 95a623bf38ead7919df17e94cadc2e7	NA	NA	–	–	–	CLI	–	...
	vEdge Cloud	b93b51c7-cd1a-4de8-6f75-bd4e2425437c	Token - d94cf702f32fbccd4f5011a6f7ca9110d	NA	NA	–	–	–	CLI	–	...
	vEdge Cloud	a96dc7b2-a269-274b-3bf2-e62351309ddd	Token - 2864efe3866763c2dbd773b542bc34c1	NA	NA	–	–	–	CLI	–	...

Activity Verification

You have completed this task when you attain these results:

- You have successfully removed a vEdge device from vManage inventory.

Lab 3: Add vEdge to vManage Inventory

In this Lab, you will add a new device to the vManage inventory.

Activity Procedure

Complete these steps:

Step 1 The following devices will be used in this Lab.

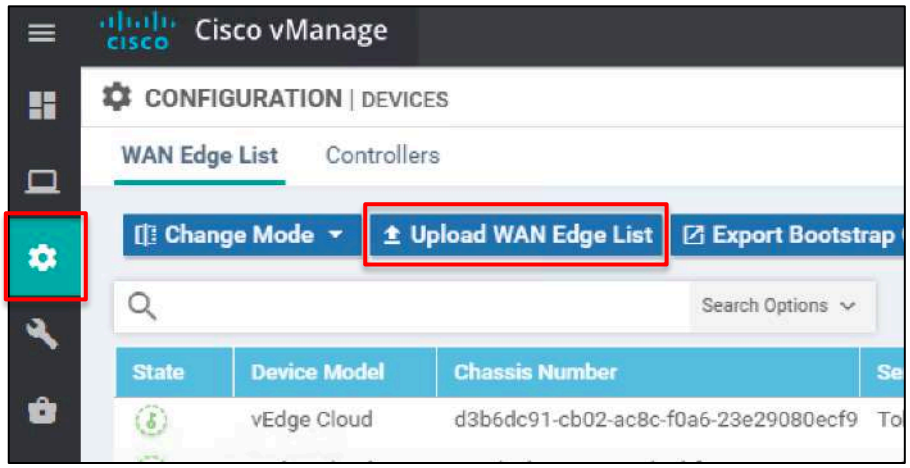
Device Name	IP Address or URL	Username	Password	LAB
vManage-1	1.4.28.28 – SSH 10.82.83.131	admin	admin	NterOne Lab
Branch-vE1	1.1.7.11 – SSH 10.82.83.111	admin	admin	NterOne Lab

Step 2 Open a browser window and log in to vManage. You will use vManage to add a device to inventory by uploading a **Device Serial List**.

Step 3 The following steps will demonstrate how to add a device to the vManage inventory

 © 2020 NterOne Corporation

Step 4 Go to **Configuration > Devices** and click on **Upload WAN Edge List** from the top left of the window.



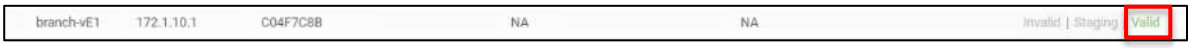
Step 5 Click the Choose File button and browse for the file name ***serialFile-v1902.viptela*** located in the Desktop > SDWAN Lab Files 19.2 folder. Check the box "Validate the uploaded vEdge List and send to controllers" and then click Upload.

Step 6 Click **OK** to confirm upload.

Step 7 The list that you are uploading contains approximately **12 devices** and only one of these devices will be newly added to the inventory. The remaining devices are **duplicates and will be ignored by vManage**. Click **OK** to confirm successful upload.

Step 8 You will be redirected to **TASK VIEW** to verify successful completion of upload and subsequent synchronization to the controllers. Inform your instructor if any task fails.

Step 9 Navigate to **Configuration > Certificates** and correctly identify the **branch-vE1**device. Click the **Valid** option link all the way to the right of the vEdge entry. Click **OK** to confirm device validation.

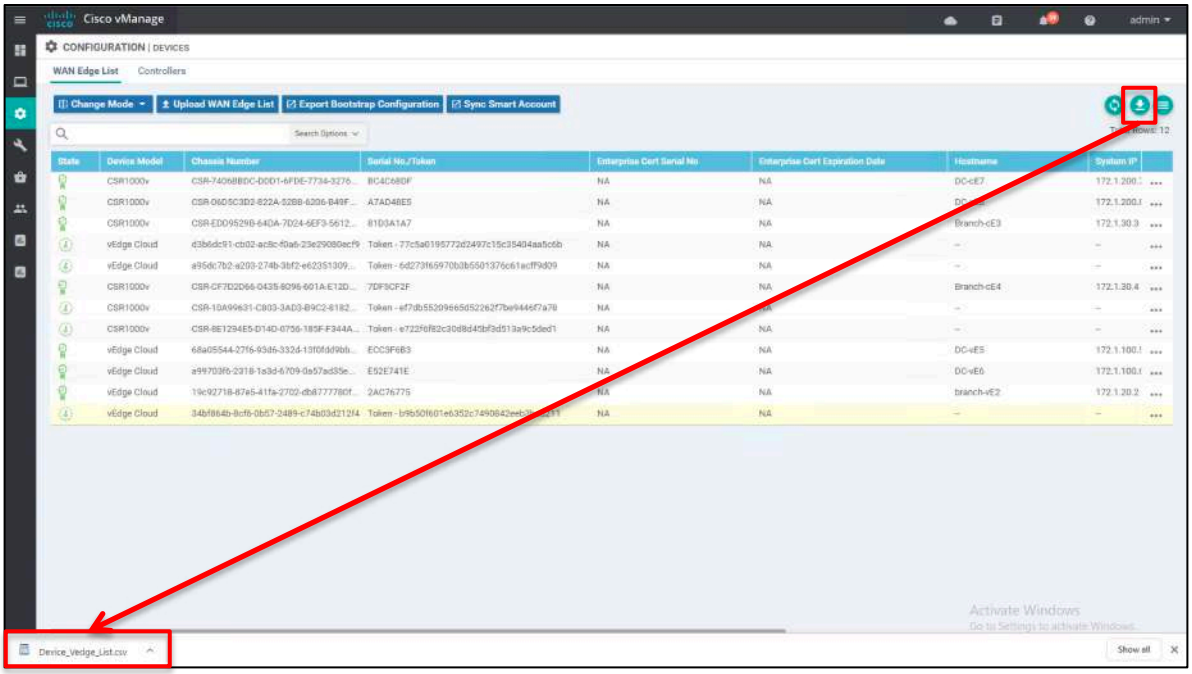


Step 10 Go to **Configuration > Devices** to identify the newly added vEdge router.

34bf864b-8cf6-0b57-2489-c74b03d212f4 **(Chassis Number)**

Step 11 **<u>Stop here</u>** and write down or copy the **Chassis and Token number** for your vEdge router. To copy the numbers, you have to click on the **Download** icon on the top right hand corner of the **Configuration > Devices** window. This will allow you to download a CSV file containing all of the device information specified within the window. This will make it easier to input the Chassis and Token number later.

 © 2020 NterOne Corporation

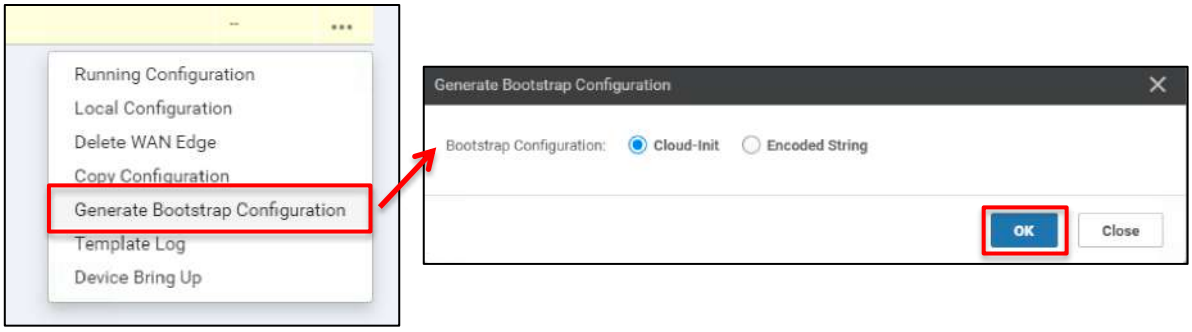


Step 12 Now that you have the **Token** you can configure your vEdge router to securely connect to the **vBond Orchestrator** and establish connectivity with the **vSmart Controllers and vManage**. The next step will be to activate the vEdge Cloud router using the Chassis and Token number.

<u>Steps 13-15 demonstrate</u> a Bootstrap Configuration which is informational only and not required in the Lab. This can be used to initialize a vEdge Cloud router during time of VM creation. For reference only!!!

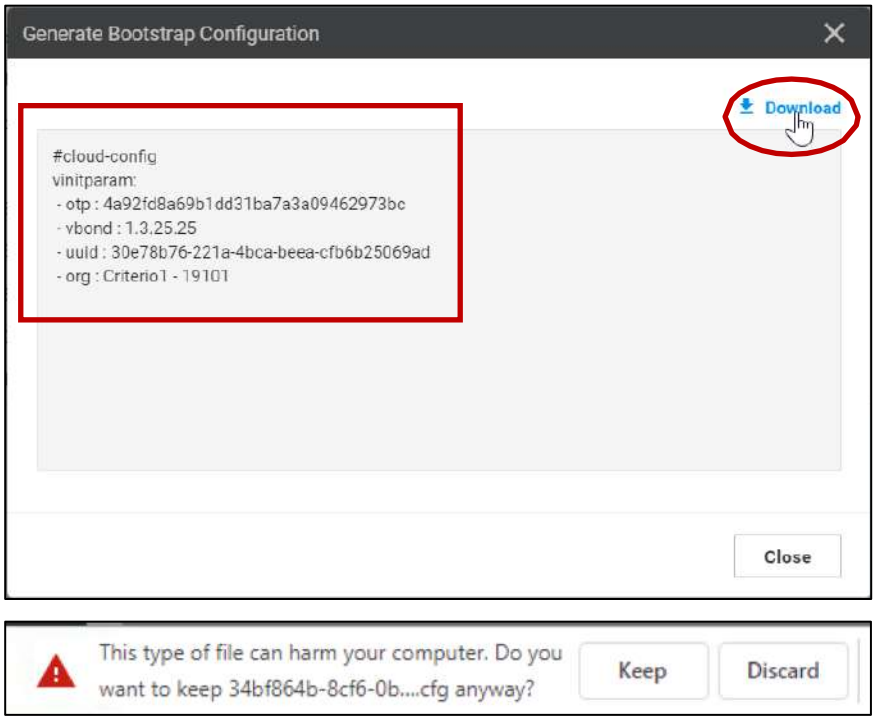
Step 13 If not already there, then go back to **Configuration > Devices** and click **…** to the right of the newly added vEdge device. Once the drop-down appears select **Generate Bootstrap Configuration.**

The bootstrap configuration can be associated with a vEdge Cloud Router VM instance and when assigned will provide an initial configuration for the deployed VM. Select **Cloud-Init** for a KVM based VM or **Encoded String** for an ESXi based VM.



Step 14 Notice that the Bootstrap configuration contains the **otp** (One-Time Password), **also known as the Token** which will be used to securely

connect to vBond. The initial configuration also contains the **vBond Host/IP**, **Organization Name**, and **uuid/Chassis Number**. Click **Download** to save the bootstrap configuration. Click **Keep** when the security warning appears and then click **Close** when done.



Step 15 You will activate the vEdge Cloud router in the next Lab **using the** CLI command "request".

Activity Verification

You have completed this task when you attain these results:

- You have successfully added a new device to the vManage inventory.

 © 2020 NterOne Corporation

Lab 4: Configure and Deploy Control-Plane

In this Lab, you will bring up the control-plane between the vEdge router and the controllers.

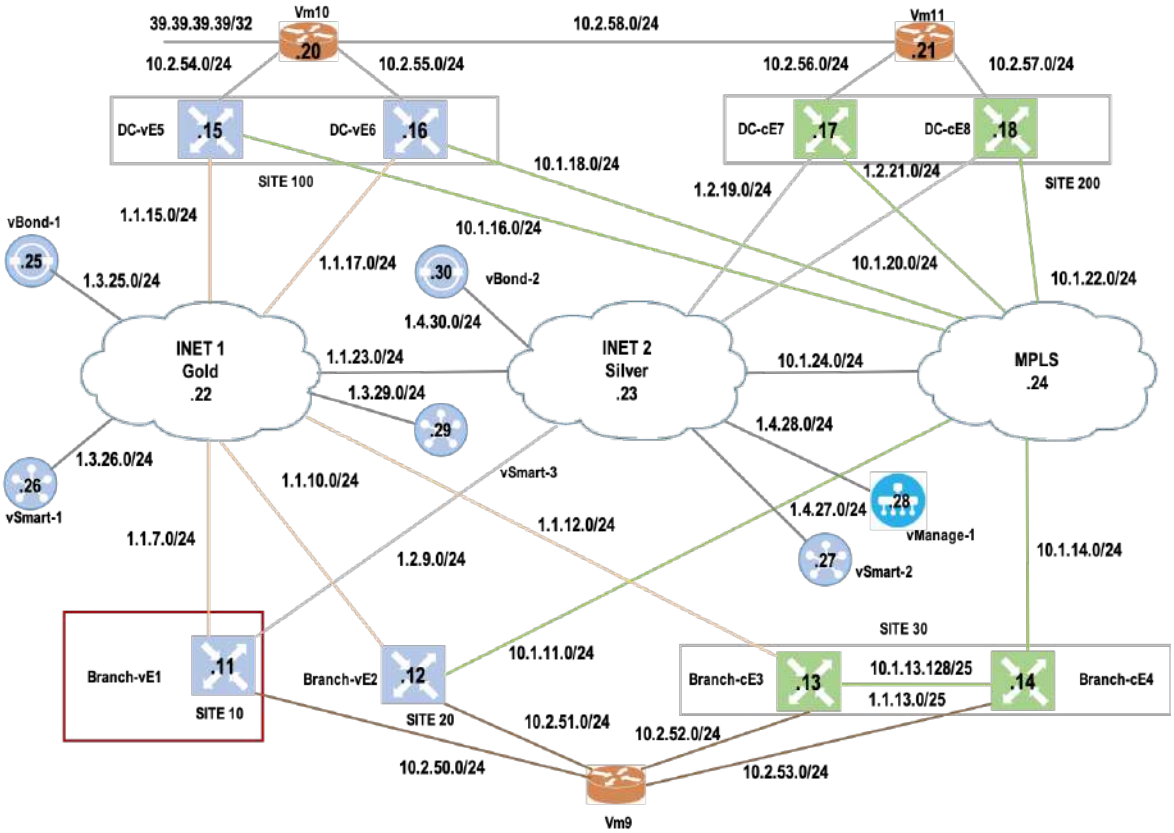
Activity Procedure

Complete these steps:

The following devices will be used in this Lab.

Device Name	IP Address or URL	Username	Password	LAB
vManage-1	https://1.4.28.28 – SSH 10.82.83.131	admin	admin	NterOne Lab
Branch-vE1	1.1.7.11 – SSH 10.82.83.111	admin	admin	NterOne Lab

Topology



Step 1 Log in to your vEdge router with credentials from the table.

Step 2 Display the vEdge control properties. Note that both the **Certificate** has not been installed and a **serial-num** (Certificate Serial number) has not been assigned.

```
branch-vE1# show control local-properties
 personality                        vedge
 sp-organization-name               NterOne Lab
 organization-name                  NterOne Lab
 root-ca-chain-status               Installed

 certificate-status                 Not-Installed
 certificate-validity               Not Applicable
 certificate-not-valid-before       Not Applicable
 certificate-not-valid-after        Not Applicable

 dns-name                           vbond.demo.local
 site-id                            10
```

 © 2020 NterOne Corporation

```
 domain-id                          1
 protocol                           dtls
 tls-port                           0
 system-ip                          172.1.10.1
 chassis-num/unique-id             34bf864b-8cf6-0b57-2489-c74b03d212f4
 serial-num                         No certificate installed
 token                              Invalid
 keygen-interval                    1:00:00:00
 retry-interval                     0:00:00:19
 no-activity-exp-interval           0:00:00:20
 dns-cache-ttl                      0:00:02:00
 port-hopped                        FALSE
 time-since-last-port-hop           0:00:00:00
 embargo-check                      success
 number-vbond-peers                 0
 number-active-wan-interfaces       0
```

Step 3 Use **show run** and verify that the configuration contains only the necessary system settings and a configured vpn 512 (for lab SSH access). Take note that **vpn 0 (WAN transport VPN)** has **not been configured and no interfaces** have been added to the configuration.

```
branch-vE1# show run
system
 host-name            branch-vE1
 system-ip            172.1.10.1
 site-id              10
 controller-group-list10
 route-consistency-check
 organization-name    NterOne Lab
 clock timezone America/Los_Angeles
 upgrade-confirm      5
 vbond vbond.demo.local

--- AAA OMITTED ---

!
omp
 shutdown
```

```
graceful-
restart
!
vpn 0
!
vpn 512
  interface
  eth0
    ip address
    10.80.88.111/20
    no shutdown
  !
!
```

Step 4 Issue the command **show interface** to verify parameters and current status. Up until now, the interfaces have not been configured for any vpn. By default, the system will display the interfaces in vpn 0.

```
branch-vE1# show interface | tab
                                                    IF       IF
                        AF                        ADMIN     OPER      ENCAP  PORT
        VPN  INTERFACE      TYPE  IP ADDRESS      STATUS    STATUS    TYPE   TYPE
               MTU -Omitted--
--------------------------------------------------------------------------------

          0  ge0/0         ipv4-                 Down      Down        null  service   1500
          0  ge0/1         ipv4-                 Down      Down        null  service   1500
          0  ge0/2         ipv4-                 Down      Down        null  service   1500
          0  ge0/3         ipv4-                 Down      Down        null  service   1500
          0  system        ipv4  172.1.10.1/32   Up        Up          null  loopback  1500

--- VPN 512 OMITTED ---
```

Step 5 Use the table below to provide the necessary values for the following steps.

Interface	IP Address	Default Next-Hop	Tunnel Color	Tunnel Encap	VPN	IPSec Authentication
ge0/1	1.1.7.11/24	1.1.7.22	gold	ipsec	0	ah-sha1-hmac sha1-hmac
ge0/2	1.2.9.11/24	1.2.9.23	silver	ipsec	0	ah-sha1-hmac sha1-hmac

Step 6 Configure **vpn 0** and assign the **interfaces** from the table provided. The default route next-hop IP addresses using the **ip route** command will be added as well. The Lab environment uses static IPs but in production it will be more likely to see DHCP-derived addresses.

 © 2020 NterOne Corporation

```
branch-vE1# config
 Entering configuration mode terminal
 branch-vE1 (config)# vpn 0
 branch-vE1(config-vpn-0)# interface ge0/1
 branch-vE1(config-interface-ge0/1)# ip address 1.1.7.11/24
 branch-vE1(config-interface-ge0/1)# no shut
 branch-vE1(config-interface-ge0/1)# exit
 branch-vE1(config-vpn-0)# interface ge0/2
 branch-vE1(config-interface-ge0/2)# ip address 1.2.9.11/24
 branch-vE1(config-interface-ge0/2)# no shut
 branch-vE1(config-interface-ge0/2)# exit
 branch-vE1(config-vpn-0)# ip route 0.0.0.0/0 1.1.7.22
 branch-vE1(config-vpn-0)# ip route 0.0.0.0/0 1.2.9.23
 branch-vE1(config-vpn-0)# show config
 vpn 0
  interface ge0/1
   ip address 1.1.7.11/24
   no shutdown
  !
  interface ge0/2
   ip address 1.2.9.11/24
   no shutdown
  !
  ip route 0.0.0.0/0 1.1.7.22
  ip route 0.0.0.0/0 1.2.9.23
 !
 branch-vE1(config-vpn-0)# commit
 Commit complete.
 branch-vE1(config-vpn-0)# end
```

Step 7 Verify that you can ping the **Default Next-hop IPs** within vpn 0. If you don't specify a **vpn** number, then the system will place you in **vpn 0 by default**. Use CTRL+C to stop a ping.

Although redundant, the example shown below uses vpn 0 for clarification.

```
branch-vE1# ping 1.1.7.22 vpn 0
 Ping in VPN 0
 PING 1.1.7.22 (1.1.7.22) 56(84) bytes of data.
 64 bytes from 1.1.7.22: icmp_seq=1 ttl=64 time=99.4 ms
 64 bytes from 1.1.7.22: icmp_seq=2 ttl=64 time=99.5 ms
 64 bytes from 1.1.7.22: icmp_seq=3 ttl=64 time=99.3 ms
 64 bytes from 1.1.7.22: icmp_seq=4 ttl=64 time=100 ms
 64 bytes from 1.1.7.22: icmp_seq=5 ttl=64 time=100 ms
 ^C
```

```
--- 1.1.7.22 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4004ms
rtt min/avg/max/mdev = 99.345/99.795/100.435/0.575 ms

branch-vE1# ping  1.2.9.23 vpn 0
Ping in VPN 0
PING 1.2.9.23 (1.2.9.23) 56(84) bytes of data.
64 bytes from 1.2.9.23: icmp_seq=1 ttl=64 time=0.167 ms
64 bytes from 1.2.9.23: icmp_seq=2 ttl=64 time=0.157 ms
64 bytes from 1.2.9.23: icmp_seq=3 ttl=64 time=0.146 ms
64 bytes from 1.2.9.23: icmp_seq=4 ttl=64 time=0.180 ms
^C
--- 1.2.9.23 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 2999ms
rtt min/avg/max/mdev = 0.146/0.162/0.180/0.017 ms

branch-vE1# show ip routes
Codes Proto-sub-type:
  IA -> ospf-intra-area, IE -> ospf-inter-area,
  E1 -> ospf-external1, E2 -> ospf-external2,
  N1 -> ospf-nssa-external1, N2 -> ospf-nssa-external2,
  e -> bgp-external, i -> bgp-internal
Codes Status flags:
  F -> fib, S -> selected, I -> inactive,
  B -> blackhole, R -> recursive

                                        PROTOCOL  NEXTHOP   NEXTHOP     NEXTHOP
 VPN    PREFIX           PROTOCOL       SUB TYPE  IF NAME   ADDR        VPN      TLOC
 ---------------------------------------------------------------------------------------
------
 0      0.0.0.0/0        static         -         ge0/2     1.2.9.23    -        -
 0      0.0.0.0/0        static         -         ge0/1     1.1.7.22    -        -
 0      1.1.7.0/24       connected      -         ge0/1     -           -        -
 0      1.2.9.0/24       connected      -         ge0/2     -           -        -
 0      172.1.10.1/32    connected      -         system    -           -        -

--- VPN 512 OMITTED ---
```

Step 8 During an earlier step, the host entry **vbond.demo.local** was configured under **system settings**. This was provisioned so that the **vEdge knows which vBond Orchestrator** to communicate with. Verify reachability to the vBond host entry.

 © 2020 NterOne Corporation

```
branch-vE1# ping vbond.demo.local
 Ping in VPN 0
 ping: vbond.demo.local: Temporary failure in name resolution
```

Step 9 It appears there is an issue resolving the hostname. Add a **DNS entry to** help resolve the IP address.

```
branch-vE1# config
 Entering configuration mode terminal
 branch-vE1(config)# vpn 0
 branch-vE1(config-vpn-0)# dns 1.4.28.89
 branch-vE1(config-vpn-0)# commit
 Commit complete.
 branch-vE1(config-vpn-0)# end
 branch-vE1# ping vbond.demo.local
 Ping in VPN 0
 PING vbond.demo.local (1.4.30.30) 56(84) bytes of data.
 64 bytes from vbond.demo.local (1.4.30.30): icmp_seq=1 ttl=62 time=127 ms
 64 bytes from vbond.demo.local (1.4.30.30): icmp_seq=2 ttl=63 time=20.5 ms
 64 bytes from vbond.demo.local (1.4.30.30): icmp_seq=3 ttl=62 time=120 ms
 64 bytes from vbond.demo.local (1.4.30.30): icmp_seq=4 ttl=63 time=28.1 ms
 ^C
 --- vbond.demo.local ping statistics ---
 5 packets transmitted, 4 received, 20% packet loss, time 4002ms
 rtt min/avg/max/mdev = 20.531/74.145/127.640/49.954 ms

branch-vE1# ping 1.4.30.30
 Ping in VPN 0
 PING 1.4.30.30 (1.4.30.30) 56(84) bytes of data.
 64 bytes from 1.4.30.30: icmp_seq=1 ttl=63 time=28.8 ms
 64 bytes from 1.4.30.30: icmp_seq=2 ttl=62 time=128 ms
 64 bytes from 1.4.30.30: icmp_seq=3 ttl=62 time=128 ms
 64 bytes from 1.4.30.30: icmp_seq=4 ttl=62 time=120 ms
 ^C
```

Step 10 Now that we have basic IP connectivity to the WAN transport segments, let's add the **tunnel interfaces** to both ge0/1 and ge0/2. The tunnel interfaces will use **ipsec** for their encapsulation. The **ipsec authentication type** has to be specified under the security settings. Colors will also be assigned to determine which interface belongs to a particular WAN transport.

```
branch-vE1# config
 Entering configuration mode terminal
 branch-vE1(config)# security
 branch-vE1(config-security)# ipsec
 branch-vE1(config-ipsec)# authentication-type ah-sha1-hmac sha1-hmac
 branch-vE1(config-ipsec)# exit
 branch-vE1(config-security)# vpn 0
 branch-vE1(config-vpn-0)# interface ge0/1
 branch-vE1(config-interface-ge0/1)# tunnel-interface
 branch-vE1(config-tunnel-interface)# encapsulation ipsec
 branch-vE1(config-tunnel-interface)# color gold
 branch-vE1(config-tunnel-interface)# exit
 branch-vE1(config-interface-ge0/1)# exit
 branch-vE1(config-vpn-0)# interface ge0/2
 branch-vE1(config-interface-ge0/2)# tunnel-interface
 branch-vE1(config-tunnel-interface)# encapsulation ipsec
 branch-vE1(config-tunnel-interface)# color silver
 branch-vE1(config-tunnel-interface)# exit
 branch-vE1(config-interface-ge0/2)# exit

branch-vE1(config-vpn-0)# show config
 vpn 0
  interface ge0/1
   tunnel-interface
    encapsulation ipsec
    color gold
    no allow-service bgp
    allow-service dhcp
    allow-service dns
    allow-service icmp
    no allow-service sshd

    no allow-service netconf
    no allow-service ntp
    no allow-service ospf
    no allow-service stun
    allow-service https
   !
  !
```

 © 2020 NterOne Corporation

```
 interface ge0/2
  tunnel-interface
   encapsulation ipsec
   color silver
   no allow-service bgp
   allow-service dhcp
   allow-service dns
   allow-service icmp
   no allow-service sshd
   no allow-service netconf
   no allow-service ntp
   no allow-service ospf
   no allow-service stun
   allow-service https
  !
 !
!
branch-vE1(config-vpn-0)# commit
Commit complete.
branch-vE1(config-vpn-0)# end
```

Step 11 Using the following commands verify that no control-plane connections have been established.

```
branch-vE1# show control connections
```

```
branch-vE1# show control local-properties
 personality                        vedge
 sp-organization-name               NterOne_Lab
 organization-name                  NterOne_Lab
 root-ca-chain-status               Installed
 certificate-status                 Not-Installed
 certificate-validity               Not Applicable
 certificate-not-valid-before       Not Applicable
 certificate-not-valid-after        Not Applicable
 dns-name                           vbond.demo.local
 site-id                            10
 domain-id                          1
 protocol                           dtls
```

```
tls-port                          0
system-ip                         172.1.10.1
chassis-num/unique-id             34bf864b-8cf6-0b57-2489-c74b03d212f4
serial-num                        No certificate installed
token                             Invalid
keygen-interval                   1:00:00:00
retry-interval                    0:00:00:19


no-activity-exp-interval          0:00:00:20
dns-cache-ttl                     0:00:02:00
port-hopped                       TRUE
time-since-last-port-hop          0:00:03:12
embargo-check                     success
number-vbond-peers                0
number-active-wan-interfaces      2


 NAT TYPE: E -- indicates End-point independent mapping
           A -- indicates Address-port dependent mapping
           N -- indicates Not learned
           Note: Requires minimum two vbonds to learn the NAT type


                          PUBLIC          PUBLIC PRIVATE         PRIVATE
PRIVATE                             MAX   RESTRICT/           LAST         SPI TIME    NAT  VM
 INTERFACE                IPv4            PORT   IPv4            IPv6
PORT    VS/VM COLOR            STATE CNTRL CONTROL/     LR/LB  CONNECTION   REMAINING   TYPE
CON

STUN                                                PRF
 ---------------------------------------------------------------------------------------------
---------------------------------------------------------------------------------------------
--------------
 ge0/1                    1.1.7.11        12346  1.1.7.11        ::
12346    0/0  gold             down   2      no/yes/no   No/No  0:02:10:12   0:09:49:47  N    5
 ge0/2                    1.2.9.11        12346  1.2.9.11        ::
12346    0/0  silver           down   2      no/yes/no   No/No  0:02:10:12   0:09:49:47  N    5


branch-vE1# show control valid-vsmarts
% No entries found.
```

 © 2020 NterOne Corporation

Step 12 The control connections have not yet been established and the vEdge router has not established a connection with either the vSmart controllers or vManage.

Step 13 Go to **Configuration > Devices** and verify that there have not been any changes since adding the device to the inventory. Once a control-plane connection has been established, vManage will have access to the vEdge router and its system parameters will be displayed in the vManage GUI.

Step 14 There's just one more thing missing. The vEdge router has to be activated with its assigned **Chassis Number** along with the **Token** that was generated on vManage.

Step 15 Use the command **request** to activate the router. This will allow it to use the one-time password (Token) to establish a secure connection with vBond. Make sure to use **CORRECT values** for the **Chassis Number and the Token** that were either written down or copied earlier.

```
branch-vE1# request vedge-cloud activate chassis-number (Verify Chassis
matches vManage Inventory) token (Verify Token matches vManage
Inventory)

branch-vE1# request vedge-cloud activate chassis-number 34bf864b-8cf6-
0b57-2489-c74b03d212f4 token b9b50f601e6352c7490842eeb3bad211
```

Step 16 After a couple of minutes, use the following commands to verify that control-plane connectivity has been established.

Note: Please keep in mind that serial numbers as the relate to cEdge and vEdge devices can vary Pod-to-Pod. As a result these show command are provided as exampls of where to look and what to look for. Some information will vary.

```
branch-vE1# show control local-properties
 personality                        vedge
 sp-organization-name               NterOne Lab
 organization-name                  NterOne Lab
 root-ca-chain-status               Installed
 certificate-status                 Installed
 certificate-validity               Valid
 certificate-not-valid-before       Oct 10 12:54:40 2019 GMT
 certificate-not-valid-after        Oct 07 12:54:40 2029 GMT
 dns-name                           vbond.demo.local
 site-id                            10
 domain-id                          1
```

```
protocol                          dtls
tls-port                          0
system-ip                         172.1.10.1
chassis-num/unique-id             34bf864b-8cf6-0b57-2489-c74b03d212f4
serial-num                        C04F7C8B
token                             Invalid
keygen-interval                   1:00:00:00

retry-interval                    0:00:00:15
no-activity-exp-interval          0:00:00:20
dns-cache-ttl                     0:00:02:00
port-hopped                       TRUE
time-since-last-port-hop          0:01:22:09
embargo-check                     success
number-vbond-peers                0
number-active-wan-interfaces      2

NAT TYPE: E -- indicates End-point independent mapping
          A -- indicates Address-port dependent mapping
          N -- indicates Not learned
          Note: Requires minimum two vbonds to learn the NAT type


                         PUBLIC          PUBLIC PRIVATE         PRIVATE
PRIVATE                            MAX   RESTRICT/           LAST         SPI TIME    NAT  VM

 INTERFACE               IPv4            PORT   IPv4            IPv6
PORT    VS/VM COLOR            STATE CNTRL CONTROL/     LR/LB  CONNECTION   REMAINING   TYPE
CON

STUN                                            PRF

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

 ge0/1                   1.1.7.11        12366  1.1.7.11        ::
12366    2/0  default          up     2      no/yes/no   No/No  0:01:18:25   0:10:40:06  E    5

 ge0/2                   1.2.9.11        12366  1.2.9.11        ::
12366    2/1  silver           up     2      no/yes/no   No/No  0:01:20:15   0:10:39:29  E    5
```

 © 2020 NterOne Corporation

```
branch-vE1# show control connections                    (Formatted for clarity)

                                                  PEER P              PEER
PEER TYPEPEER PEER SYSTEM SITE   DOMAIN PEER PRIVATEPORT  PEER PU      PUB      LOCAL  STATE
                                                                IP
-----------------------------------------------------------------------------------------
vsmart              172.1     1      1.3.26.26        1.3.26.26                gold   up
vsmart              172.1     1      1.3.26.26        1.3.26.26                silver up
vsmart              172.1     1      1.3.29.29        1.3.29.29                gold   up
vsmart              172.1     1      1.3.29.29        1.3.29.29                silver up
vmanage             172.1     0      1.4.28.28        1.4.28.28                gold   Up
```

```
branch-vE1# show control valid-vsmarts

 SERIAL NUMBER     ORG
 -------------------------------
 8D86CCAF08DA3D1C  NterOne Lab
 8D86CCAF08DA3D2D  NterOne Lab
 8D86CCAF08DA3D2E  NterOne Lab
 8D86CCAF08DA3D2F  NterOne Lab
```

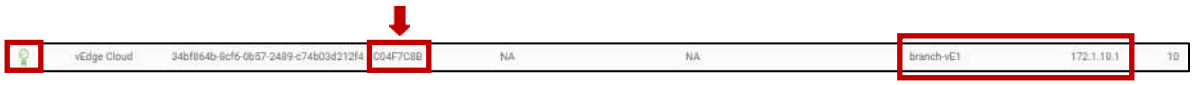
```
branch-vE1# show control valid-vmanage-id

 CHASSIS NUMBER
 --------------------------------------
 249dbae3-f45f-49a8-b3ef-178d17ac9a2d
 618c3989-5dbe-4f92-95b8-cca81d1a466e
```

```
branch-vE1# show interface | tab

                                          IF      IF      IF
                 AF                       ADMIN   OPER    TRACKER  ENCAP
 VPN  INTERFACE  TYPE  IP ADDRESS         STATUS  STATUS  STATUS   TYPE   PORT TYPE  MTU
 ---------------------------------------------------------------------------------------
 0    ge0/0      ipv4  -                 Down    Down    NA       null   service    1500
 0    ge0/1      ipv4  1.1.7.11/24        Up      Up      NA       null   transport  1500
 0    ge0/2      ipv4  1.2.9.11/24        Up      Up      NA       null   transport  1500
 0    ge0/3      ipv4  -                 Down    Down    NA       null   service    1500
 0    system     ipv4  172.1.10.1/32      Up      Up      NA       null   loopback   1500

--- VPN 512 OMITTED ---
```

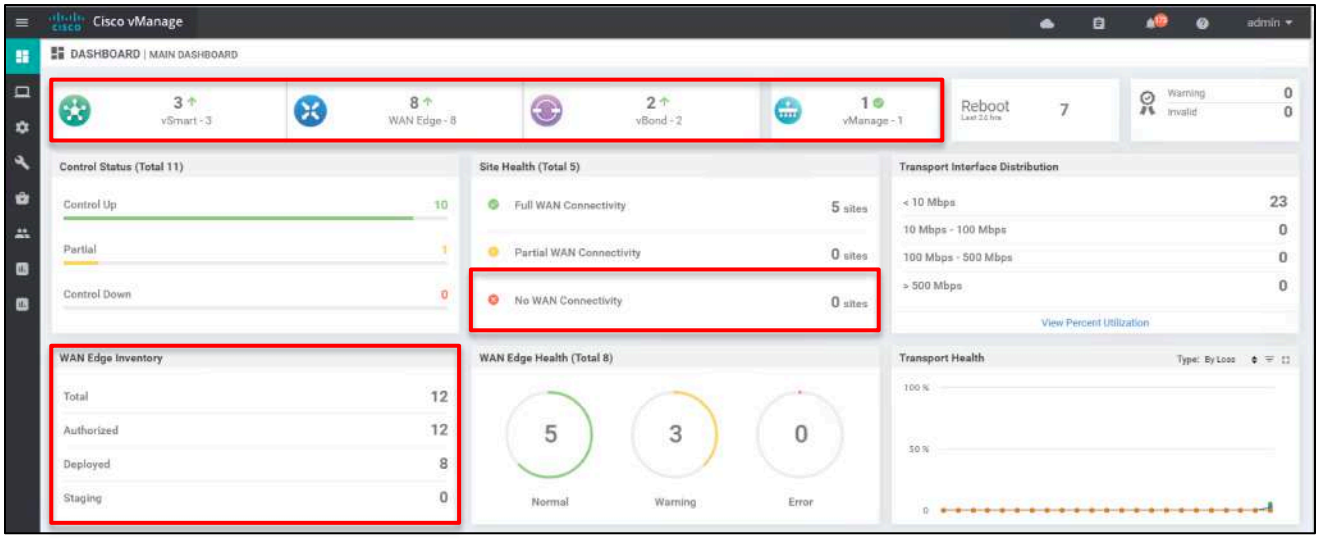
```
branch-vE1# show certificate installed

 Installed device certificates
 -----------------------------
 Certificate:
     Data:
         Version: 1 (0x0)
         Serial Number: 3226434699 (0xc04f7c8b)
     Signature Algorithm: sha256WithRSAEncryption
         Issuer: C=US, CN=618c3989-5dbe-4f92-95b8-cca81d1a466e,
O=Viptela
         Validity
             Not Before: Oct 10 12:54:40 2019 GMT
             Not After : Oct  7 12:54:40 2029 GMT
         Subject: C=US, ST=California, L=San Jose, OU=NterOne Lab,
O=Viptela LLC, CN=vedge-34bf864b-8cf6-0b57-2489-c74b03d212f4-
3.viptela.com/emailAddress=support@viptela.com


--- OUTPUT OMITTED ---
```

Step 17 It appears that the vEdge router has established control-plane connectivity.

Step 18 In vManage, go to **Configuration > Devices**. Identify the correct vEdge device and verify that a connection to the vEdge router has been established. Notice that a Serial Number has been assigned (arrow).



Step 19 Go back to the vManage Dashboard to verify the health of the system and determine that all devices are present within the network.



 © 2020 NterOne Corporation

Note	The **"No Connectivity 1 sites"** entry under 'Site Health View' will be resolved in the next lab.

Activity Verification

You have completed this task when you attain these results:

- You have successfully established control-plane connectivity between the vEdge router and the controllers.

Lab 5: Configure and Deploy an Overlay Network

In this Lab, you will bring up the **Service** Overlay on VPN 1

Activity Procedure

Complete these steps:

Step 1 The following devices will be used in this Lab.

Device Name	IP Address or URL	Username	Password	LAB
vManage-1	https://1.4.28.28 – SSH 10.82.83.131	admin	admin	NterOne Lab
Branch-vE1 vm9 DC-vE5	1.1.7.11 – SSH 10.82.83.111 SSH 10.82.83.141 SSH 10.82.83.121	admin admin admin	admin admin admin	NterOne Lab NterOne Lab NterOne Lab

Interface	IP Address	VPN	OSPF	Router-id
ge0/0	10.2.50.11/24	1	Area 0	172.1.10.1
loopback 1	99.99.99.99/32	1	Area 0	Passive-interface

Step 2 Log in to your vEdge router with credentials from the table.

Step 3 Display the routes on the vEdge router

```
branch-vE1# show ip routes                          (Formatted for clarity)
 Codes Proto-sub-type:
   IA -> ospf-intra-area, IE -> ospf-inter-area,
   E1 -> ospf-external1, E2 -> ospf-external2,
   N1 -> ospf-nssa-external1, N2 -> ospf-nssa-external2,
   e -> bgp-external, i -> bgp-internal
 Codes Status flags:
   F -> fib, S -> selected, I -> inactive,
   B -> blackhole, R -> recursive
```

 © 2020 NterOne Corporation

```
                                PROTOCOL  NEXTHOP     NEXTHOP   NEXTHOP
VPN   PREFIX        PROTOCOL    SUB TYPE  IF NAME     ADDR      VPN     TLOC IP  COLOR
-----------------------------------------------------------------------------------------
0     0.0.0.0/0     static      -         ge0/1       1.1.7.22  -       -        -
0     0.0.0.0/0     static      -         ge0/2       1.2.9.23  -       -        -
0     1.1.7.0/24    connected   -         ge0/1       -         -       -        -
0     1.2.9.0/24    connected   -         ge0/2       -         -       -        -
0     172.1.10.1/32 connected   -         system      -         -       -        -
--- VPN 512 OMITTED ---
```

Step 4 The only routes that are available at this point are routes responsible for WAN Transport connectivity – **VPN 0**. **Internal prefixes are not available as of yet due to the absence of a Service VPN**. VPN 0 and VPN 512 are used for transport and management; whereas creating other vpns such as 1 is needed in order to establish an Overlay.

Step 5 For an Overlay network to exchange routing prefixes, another important control-plane protocol is required. OMP needs to be enabled and configured so that routing information can be exchanged with other vEdge devices across the SD-WAN.

Step 6 Issue the following commands to verify Overlay functionality.

```
branch-vE1# show omp peers


branch-vE1# show omp routes
branch-vE1# show omp tlocs
  % No entries found.
branch-vE1# show omp summary
oper-state              DOWN
admin-state             DOWN
personality             vedge


--- OUTPUT OMITTED ---


branch-vE1# show run omp
omp
  shutdown
  graceful-restart
!
branch-vE1#
```

Step 7 Use the command **no shut** under OMP settings to enable OMP. Once activated, OMP should establish a connection with the vSmart controllers, although no routes are being received or sent.

```
branch-vE1(config)# omp
branch-vE1(config-omp)# no shut
branch-vE1(config-omp)# commit
Commit complete.
branch-vE1(config-omp)# end
branch-vE1# show run omp
omp
   no shutdown
   graceful-restart
!
branch-vE1# show omp peers    (Output may vary based on Software
Version)
R -> routes received
I -> routes installed
S -> routes sent

                        DOMAIN    OVERLAY   SITE
PEER            TYPE    ID        ID        ID        STATE    UPTIME          R/I/S

-
          ---------------------------------------------------------------------
          -----------------
172.1.0.16      vsmart  1         1         1011      up       0:00:00:11      0/0/0
172.1.0.19      vsmart  0         1         0         init                     0/0/0
                                     OR

172.1.0.16      vsmart  1         1         1011      up       0:00:00:22      0/0/0
172.1.0.17      vsmart  1         1         1012      up       0:00:00:21      0/0/0
```

Step 8 To establish a before and after from a routing perspective, try pinging the Loopback interface on Vm10 (39.39.39.39). The ping should **fail** because there is no route to this network. Recall the default vpn **VPN 0** will be used as the basis for the ping.

```
branch-vE1# ping 39.39.39.39
Ping in VPN 0
PING 39.39.39.39 (39.39.39.39) 56(84) bytes of data.
^C
--- 39.39.39.39 ping statistics ---
4 packets transmitted, 0 received, 100% packet loss, time 3000ms
```

Step 9 **Create VPN 1** and assign the interfaces listed in the Table provided earlier. This "Service" vpn will be used as the Overlay network on the vEdge router.

 © 2020 NterOne Corporation

```
branch-vE1# config
branch-vE1(config)# vpn 1
branch-vE1(config-vpn-1)# interface ge0/0
branch-vE1(config-interface-ge0/0)# ip address 10.2.50.11/24
branch-vE1(config-interface-ge0/0)# no shut
branch-vE1(config-interface-ge0/0)# exit
branch-vE1(config-vpn-1)# interface loopback1
branch-vE1(config-interface-loopback1)# ip address 99.99.99.99/32
branch-vE1(config-interface-loopback1)# no shut
branch-vE1(config-interface-loopback1)# commit
Commit complete.
branch-vE1(config-interface-loopback1)# end
branch-vE1#
branch-vE1# show run vpn 1
vpn 1
 interface ge0/0
  ip address 10.2.50.11/24
  no shutdown
 !
 interface loopback1
  ip address 99.99.99.99/32
  no shutdown
 !
!
```

Step 10 Now that the Overlay VPN 1 has been configured, use the command **show omp peers** to display the OMP peers that have been established. Determine how many routes have been **received and sent**.

```
branch-vE1# show omp peers                (Output may vary)
R -> routes received
I -> routes installed
S -> routes sent
```

PEER	TYPE	DOMAIN ID	OVERLAY ID	SITE ID	STATE	UPTIME	
172.1.0.16	vsmart	1	1	1011	up	0:00:39:21	30/15/0
172.1.0.19	vsmart	0	1	0	init		0/0/0

OR

172.1.0.16	vsmart	1	1	1011	up	0:00:04:10	18/9/0
172.1.0.17	vsmart	1	1	1012	up	0:00:04:09	24/6/0

```
branch-vE1# show ip routes                          (Formatted for clarity)
 Codes Proto-sub-type:
   IA -> ospf-intra-area, IE -> ospf-inter-area,
   E1 -> ospf-external1, E2 -> ospf-external2,
   N1 -> ospf-nssa-external1, N2 -> ospf-nssa-external2,
   e -> bgp-external, i -> bgp-internal
 Codes Status flags:
   F -> fib, S -> selected, I -> inactive,
   B -> blackhole, R -> recursive

                                          PROTOCOL  NEXTHOP NEXTHOP   NEXTHOP
 VPN    PREFIX        PROTOCOL  SUB TYPE  IF NAME ADDR      VPN   TLOC IP     COLOR ENCAP
STATUS
 ---------------------------------------------------------------------------------------------
-----
 0      0.0.0.0/0     static    -         ge0/1   1.1.7.22  -     -           -     -      F,S
 0      0.0.0.0/0     static    -         ge0/2   1.2.9.23  -     -           -     -      F,S
 0      1.1.7.0/24    connected -         ge0/1   -         -     -           -     -      F,S
 0      1.2.9.0/24    connected -         ge0/2   -         -     -           -     -      F,S
 0      172.1.10.1/32 connected -         system  -         -     -           -     -      F,S
 1      0.0.0.0/0     omp       -         -       -         -     172.1.100.5 gold ipsec   F,S
 1      0.0.0.0/0     omp       -         -       -         -     172.1.100.6 gold ipsec   F,S
 1      10.2.50.0/24  connected -         ge0/0   -         -     -           -    -       F,S
 1      10.2.51.0/24  omp       -         -       -         -     172.1.20.2  gold ipsec   F,S
 1      10.2.54.0/24  omp       -         -       -         -     172.1.100.5 gold ipsec   F,S
 1      10.2.55.0/24  omp       -         -       -         -     172.1.100.6 gold ipsec   F,S
 1      10.2.58.0/24  omp       -         -       -         -     172.1.100.5 gold ipsec   F,S
 1      10.2.58.0/24  omp       -         -       -         -     172.1.100.6 gold ipsec   F,S
 1      39.39.39.39/32omp       -         -       -         -     172.1.100.5 gold ipsec   F,S
 1      39.39.39.39/32omp       -         -       -         -     172.1.100.6 gold ipsec   F,S
 1      99.99.99.99/32connected -         loopback1   -     -     -           -    -       F,S
```

--- *VPN 512 OMITTED* ---

Step 11 Try to ping loopback 1 on Vm10. It should be unsuccessful, because the local prefixes have not been distributed into OMP and hence not available. To confirm, verify if any of the local VPN 1 prefixes are present on DC-vE5 which is located across the WAN. **Open another putty session to DC-vE5.**

```
branch-vE1# ping 39.39.39.39 vpn 1
Ping in VPN 1
PING 39.39.39.39 (39.39.39.39) 56(84) bytes of data.
^C
--- 39.39.39.39 ping statistics ---
8 packets transmitted, 0 received, 100% packet loss, time 6999ms

DC-vE5# show ip route 10.2.50.0/24        (View from DC-vE5)
Codes Proto-sub-type:
  IA -> ospf-intra-area, IE -> ospf-inter-area,
  E1 -> ospf-external1, E2 -> ospf-external2,
  N1 -> ospf-nssa-external1, N2 -> ospf-nssa-external2,
  e -> bgp-external, i -> bgp-internal
Codes Status flags:
F -> fib, S -> selected, I -> inactive,
```

 © 2020 NterOne Corporation

```
        B -> blackhole, R -> recursive

                                                PROTOCOL  NEXTHOP     NEXTHOP
NEXTHOP
VPN     PREFIX                PROTOCOL          SUB TYPE  IF NAME     ADDR            VPN
TLOC IPCOLOR                  ENCAP  STATUS
-----------------------------------------------------------------------------------------

     DC-vE5# show ip route 99.99.99.99/32        (View from DC-vE5)
     Codes Proto-sub-type:
       IA -> ospf-intra-area, IE -> ospf-inter-area,
       E1 -> ospf-external1, E2 -> ospf-external2,
       N1 -> ospf-nssa-external1, N2 -> ospf-nssa-external2,
       e -> bgp-external, i -> bgp-internal
     Codes Status flags:
       F -> fib, S -> selected, I -> inactive,
       B -> blackhole, R -> recursive

                                                PROTOCOL  NEXTHOP     NEXTHOP
NEXTHOP
VPN     PREFIX                PROTOCOL          SUB TYPE  IF NAME     ADDR            VPN
TLOC IPCOLOR                  ENCAP  STATUS
-----------------------------------------------------------------------------------------
```

Step 12 Configure OMP to **advertise the connected routes** that were configured in VPN 1 (ge0/0 and loopback1).

```
branch-vE1(config)# omp
branch-vE1(config-omp)# advertise connected
branch-vE1(config-omp)# commit
Commit complete.
branch-vE1(config-omp)# end
branch-vE1# show run omp
omp
 no shutdown
 graceful-restart
 advertise connected
!
```

Step 13 Issue the following commands to verify connectivity. Notice that there are now **routes being sent** using OMP. Confirm reachability across the WAN and ping the addresses from before.

```
branch-vE1# show omp peers              (Output may vary)
R -> routes received
I -> routes installed
S -> routes sent
```

```
172.1.0.16      vsmart   1       1          1011       up     0:00:25:30        18/9/4
172.1.0.17      vsmart   1       1          1012       up     0:00:25:29        24/6/4
```

```
branch-vE1# show omp tlocs | tab (| tab - optional) (Output may vary)
C  -> chosen
I  -> installed
Red -> redistributed
Rej -> rejected
L  -> looped
R  -> resolved
S  -> stale
Ext -> extranet
Inv -> invalid

 ADDRESS                                                                    --COLUMNS--
FAMILY    TLOC IP          COLOR          ENCAP   FROM PEER          STATUS   --OMITTED--
-----------------------------------------------------------------------------
ipv4      172.1.10.1       gold           ipsec   0.0.0.0            C,Red,R
          172.1.10.1       silver         ipsec   0.0.0.0            C,Red,R
          172.1.20.2       mpls           ipsec   172.1.0.17         C,I,R
          172.1.20.2       gold           ipsec   172.1.0.17         C,I,R
          172.1.30.3       mpls           ipsec   172.1.0.16         C,I,R
                                                   172.1.0.17         C,R
          172.1.30.3       gold           ipsec   172.1.0.16         C,I,R
                                                   172.1.0.17         C,R
          172.1.30.4       mpls           ipsec   172.1.0.16         C,I,R
                                                   172.1.0.17         C,R
          172.1.30.4       gold           ipsec   172.1.0.16         C,I,R
                                                   172.1.0.17         C,R
          172.1.100.5      mpls           ipsec   172.1.0.16         C,I,R
                                                   172.1.0.17         C,R
          172.1.100.5      gold           ipsec   172.1.0.16         C,I,R
                                                   172.1.0.17         C,R
          172.1.100.6      mpls           ipsec   172.1.0.16         C,I,R
          172.1.100.6      gold           ipsec   172.1.0.16         C,I,R

 --- OUTPUT OMITTED ---

branch-vE1# show omp summary                    (Output may vary)
oper-state            UP
admin-state           UP
```

 © 2020 NterOne Corporation

```
personality            vedge
omp-uptime             0:06:23:14
routes-received        34
routes-installed       15
routes-sent            4
tlocs-received         16
tlocs-installed        14
tlocs-sent             2
services-received      2
services-installed     0
services-sent          2

branch-vE1# ping 39.39.39.39 vpn 1
Ping in VPN 1
PING 39.39.39.39 (39.39.39.39) 56(84) bytes of data.
64 bytes from 39.39.39.39: icmp_seq=1 ttl=63 time=59.8 ms
64 bytes from 39.39.39.39: icmp_seq=2 ttl=63 time=60.8 ms
64 bytes from 39.39.39.39: icmp_seq=3 ttl=63 time=62.0 ms
^C

DC-vE5# show ip route 99.99.99.99/32      (View from DC-vE5)
Codes Proto-sub-type:
  IA -> ospf-intra-area, IE -> ospf-inter-area,
  E1 -> ospf-external1, E2 -> ospf-external2,
  N1 -> ospf-nssa-external1, N2 -> ospf-nssa-external2,
  e -> bgp-external, i -> bgp-internal
Codes Status flags:
  F -> fib, S -> selected, I -> inactive, B -> blackhole, R -> recursive

                    PROTOCOL
                     PROTOCOL SUB      NEXTHOP    NEXTHOP NEXTHOP
VPN   PREFIX         TYPE             IF NAME    ADDR    VPN   TLOC IP    COLOR ENCAP STATUS
-----------------------------------------------------------------------------------------
1     99.99.99.99/   omp       -        -        -       -     172.1.10.1 gold   ipsec  F,S
          32
1     99.99.99.99/   omp       -        -        -       -     172.1.10.1 silver ipsec  F,S
          32

DC-vE5#             show ip route 10.2.50.0/24      (View from DC-vE5)

                    PROTOCOL
                     PROTOCOL SUB      NEXTHOP    NEXTHOP NEXTHOP
VPN   PREFIX         TYPE             IF NAME    ADDR    VPN   TLOC IP    COLOR ENCAP STATUS
-----------------------------------------------------------------------------------------
1     10.2.50.0/24   omp       -        -        -       -     172.1.10.1 gold   ipsec  F,S
1     10.2.50.0/24   omp       -        -        -       -     172.1.10.1 silver ipsec  F,S
```

Step 14 The internal network on the **Branch side uses OSPF to exchange routes** with other vEdge routers. Configure OSPF under VPN 1 so that your Pod's vEdge router can participate as well. Make sure to **redistribute OMP** into OSPF.

```
branch-vE1(config)# vpn 1
branch-vE1(config-vpn-1)# router
branch-vE1(config-router)# ospf
branch-vE1(config-ospf)# router-id 172.1.10.1
branch-vE1(config-ospf)# redistribute omp
branch-vE1(config-ospf)# redistribute connected
branch-vE1(config-ospf)# area 0
branch-vE1(config-area-0)# interface ge0/0
branch-vE1(ospf-if-ge0/0)# exit
branch-vE1(config-area-0)# interface loopback1
branch-vE1(config-if-loopback1)# passive-interface
branch-vE1(config-if-loopback1)# exit
branch-vE1(config-area-0)# show config

vpn 1
 router
  ospf
   router-id 172.1.10.1
   timers spf 200 1000 10000
   redistribute omp
   redistribute connected
   area 0
    interface ge0/0
    exit
    interface loopback1
     passive-interface
    exit
   exit
  !
 !
!
branch-vE1(config)# commit
Commit complete.
```

 © 2020 NterOne Corporation

```
branch-vE1(config)# end

branch-vE1# show ospf neighbor
DBsmL -> Database Summary List
RqstL -> Link State Request List
RXmtl -> Link State Retransmission List
     SOURCE       DEAD
VPN  IP ADDRESS   INTERFACE    ROUTER ID  STATE   PRIORITY    TIMER     DBsmL  RqstL  RXmtL
-----------------------------------------------------------------------------------------
1    10.2.50.19   ge0/0        172.1.90.9 full    1   35       0         0      0

branch-vE1# show ospf routes

     ROUTE                                                DEST                IF
VPN  TYPE     PREFIX         ID  AREA  COST  PATH TYPE   TYPE     NEXT HOP    NAME
-----------------------------------------------------------------------------------------
1    router   172.1.90.9/32  0   0     10    intra-area  router   10.2.50.19  ge0/0
1    network  10.2.50.0/24   0   0     10    intra-area  network  0.0.0.0     ge0/0
1    network  99.99.99.99/32 0   0      1    intra-area  network  0.0.0.0     loopback1
```

Step 15 **Advertise OSPF into OMP** and advertise any **Static routes** that are present in VPN 1.

```
branch-vE1(config)# omp
branch-vE1(config-omp)# advertise ospf external
branch-vE1(config-omp)# advertise connected
branch-vE1(config-omp)# advertise static
branch-vE1(config-omp)# commit
Commit complete.

branch-vE1(config-omp)# end
branch-vE1#

branch-vE1# show run omp
omp
 no shutdown
 graceful-restart
 advertise ospf external
```

```
 advertise connected
 advertise static
!
```

Step 16 **Verify that prefix** 99.99.99.99/32 is advertised **to** vm9 using OSPF. Open a third putty session to vm9 at 10.82.83.141

```
vm9# sh ip route 99.99.99.99/32
Codes Proto-sub-type:
  IA -> ospf-intra-area, IE -> ospf-inter-area,
  E1 -> ospf-external1, E2 -> ospf-external2,
  N1 -> ospf-nssa-external1, N2 -> ospf-nssa-external2,
  e -> bgp-external, i -> bgp-internal
Codes Status flags:
  F -> fib, S -> selected, I -> inactive,
  B -> blackhole, R -> recursive

                          PROTOCOL NEXTHOP NEXTHOP          NEXTHOP
VPN          PREFIX       PROTOCOL SUB TYPE IF NAME ADDR    VPN  TLOC IP COLOR ENCAP STAT
-------------------------------------------------------------------------------
1    99.99.99.99/32  ospf     IA       ge0/0   10.2.50.11  -    -      -    -     F,S
2    99.99.99.99/32  ospf     E2       ge0/1   10.2.51.12  -    -      -    -     F,S
3    99.99.99.99/32  ospf     E2       ge0/2   10.2.53.14  -    -      -    -     F,S
```

```
vm9#        show ospf routes vpn 1 | tab

     ROUTE                                                  DEST                IF
VPN  TYPE      PREFIX          ID  AREA  COST  PATH TYPE    TYPE     NEXT HOP   NAME
-----------------------------------------------------------------------------------------
1    router    172.1.10.1/32   0   0     10    intra-area   router   10.2.50.11 ge0/0
1    network   10.2.50.0/24    0   0     10    intra-area   network  0.0.0.0    ge0/0
1    network   99.99.99.99/32  0   0     11    intra-area   network  10.2.50.11 ge0/0
1    external  10.2.51.0/24    0   -     -     external2    network  10.2.50.11 ge0/0
1    external  10.2.51.128/25  0   -     -     external2    network  10.2.50.11 ge0/0
```

--- OUTPUT OMITTED ---

 © 2020 NterOne Corporation

```
branch-vE1# sh run vpn 1
vpn 1
 router
  ospf
   router-id 172.1.10.1
   timers spf 200 1000 10000
   redistribute omp
   redistribute connected
  area 0
   interface ge0/0
   exit
   interface loopback1
    passive-interface
   exit
  exit
 !
!
interface ge0/0
 ip address 10.2.50.11/24
 no shutdown
!
interface loopback1
  ip address 99.99.99.99/32
  no shutdown
 !
!
```

Activity Verification

You have completed this task when you attain these results:

- You have successfully configured and verified OMP functionality
- You have successfully configured and verified OSPF routing

Lab 6: Exploring vManage GUI

In this Lab, you will explore the vManage GUI.

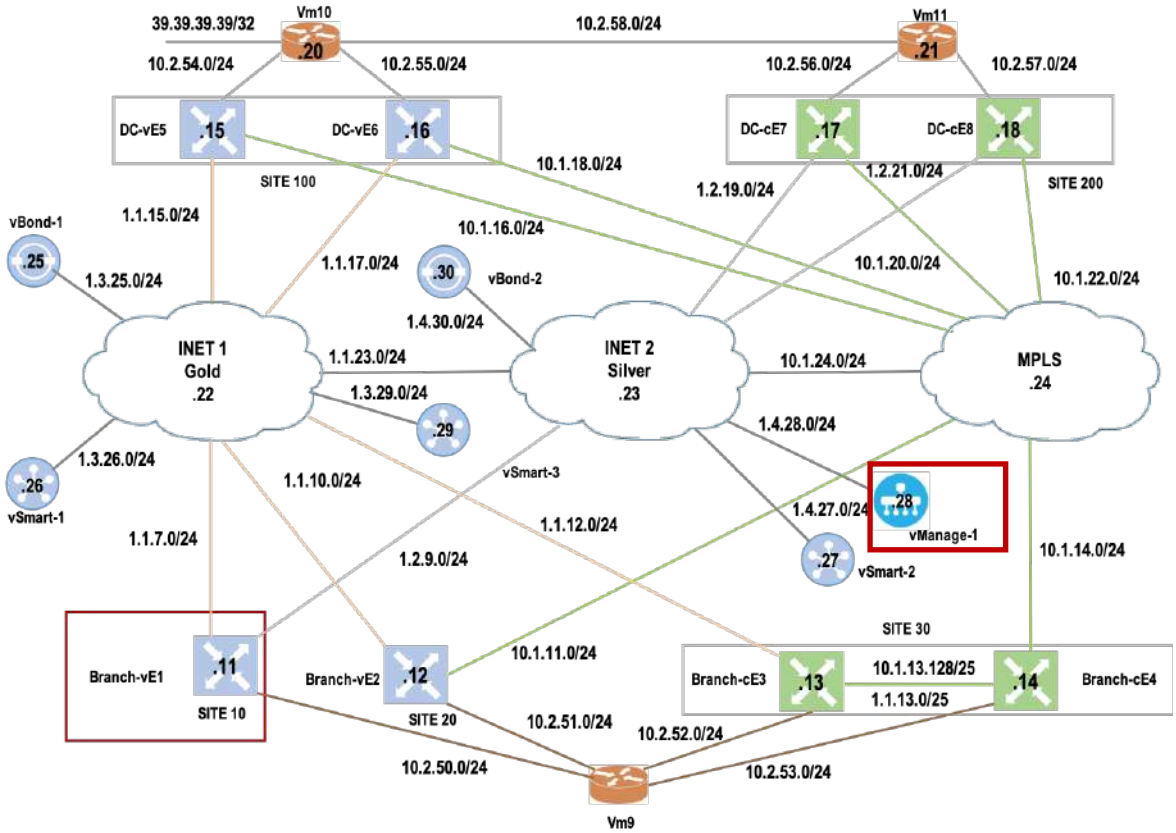
Activity Objectives

In this activity, you will become acquainted with vManage.

Device Name	IP Address or URL	Username	Password	LAB
Branch-vE1	1.1.7.11 – SSH 10.82.83.111	admin	admin	NterOne Lab
vManage-1 vBond-1	https://1.4.28.28 – SSH 10.82.83.131 SSH 10.82.83.132	admin admin	admin admin	NterOne Lab NterOne Lab

 © 2020 NterOne Corporation

Topology

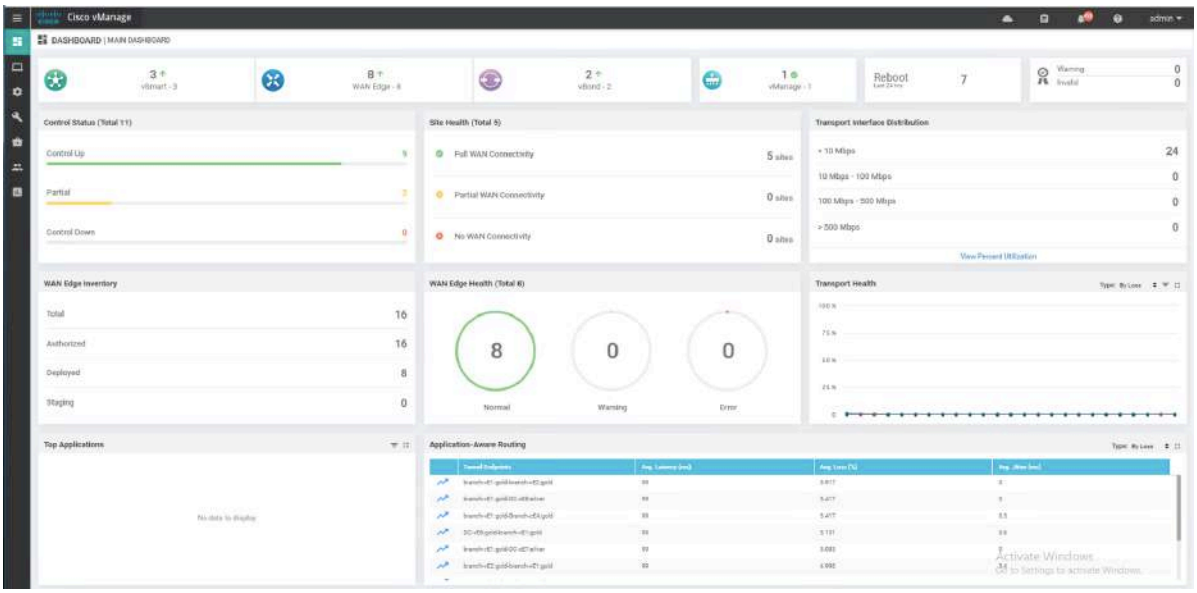


Activity Procedure

Complete these steps:

Step 1 Open a browser window and **login to vManage** (admin/admin).

Step 2 Once logged in, you'll be redirected to the **vManage Dashboard**.



Step 3 **The following steps will outline the various areas displayed on the Dashboard.**

Step 4 **The top bar** is located at the top of the vManage screen and includes the following screen elements.

Menu: Click the icon to expand or collapse the vManage menu. The vManage menu is closed by default.

CloudExpress: Enables CloudExpress service to optimize access to cloud applications. When CloudExpress service is enabled, the icon turns blue.

Tasks: Click on the icon to see a list of all active and completed tasks started from the vManage NMS. While the task is in progress, the Tasks tab displays a counter on the top. When the task is completed, the count disappears and Completed Tasks count is incremented. To view details about any task, click the task to display its Status screen.

Alarm bell: Displays the total count of all active alarms. Click on the icon to see a list of all active and cleared alarms. To view details about any alarm,

 © 2020 NterOne Corporation

click the alarm to display its Alarms Details screen.

Help: Links to product help, software version information about the vManage NMS software, and current time and time zone on the vManage server.

User profile: Click to sign out or edit user-related options in your profile.

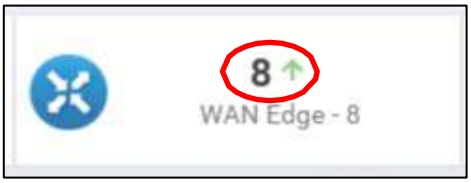
Step 5 The Device pane, which runs across the top of the Dashboard screen, displays all control connections from the vManage NMS to the vSmart controllers, vEdge routers, and vBond orchestrators in the overlay network.

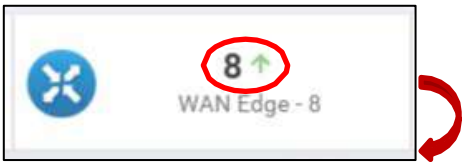
It also displays the status of the vManage NMSs in the network

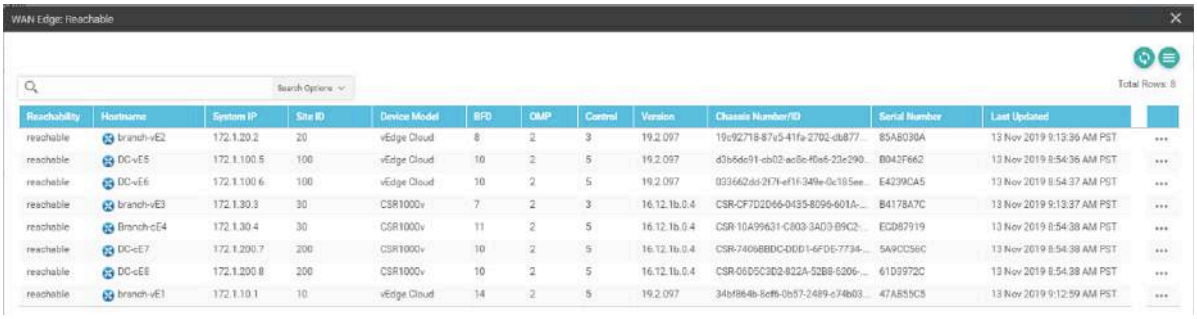
For each device, the Device pane shows:

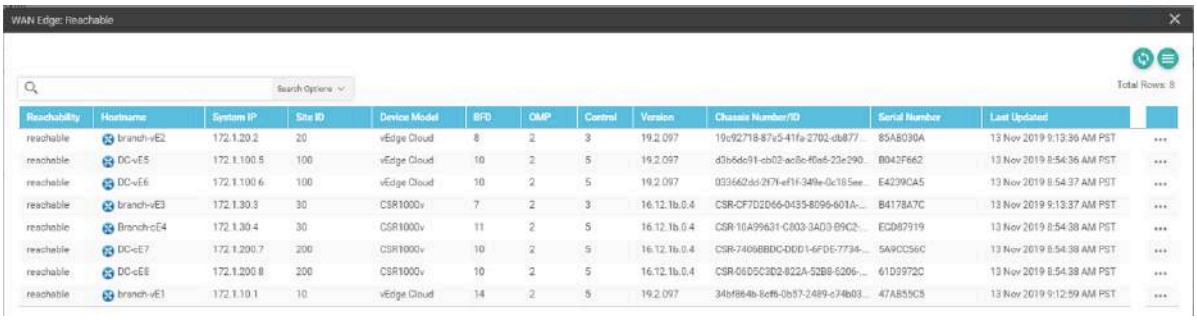
- Total number of connections
- Number of up connections
- Number of down connections

Step 6 Click the **number or the Up or Down arrow** to display a table with detailed information for each connection.



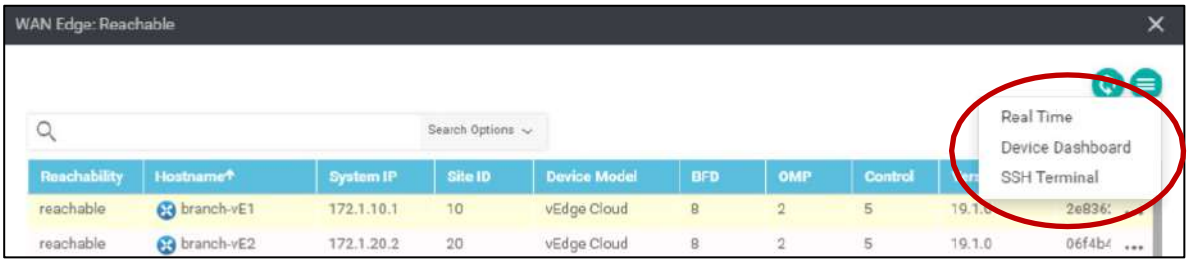




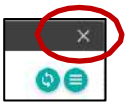


Step 7 Click the More Actions icon **…** to the right of each table row to access the Device Dashboard or Real Time view in the **Monitor > Network** screen or to access the **Tools > SSH Terminal** screen.

Real Time and SSH Terminal will be covered in more detail later in the Lab.



Step 8 **Close the Table** when done viewing connections.



Step 9 **The Reboot pane** displays the total number of reboots in the last 24 hours **for all devices in the network**, including soft and cold reboots and reboots that occurred as a result of power-cycling a device

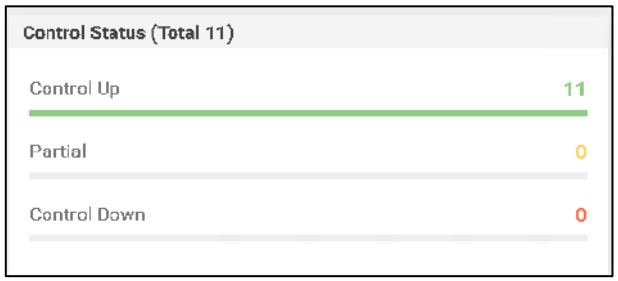
Step 10 **The Certificates pane** displays the state of all certificates on **all controller devices**, and it shows a count of all expired or invalidated certificates.

Step 11 **The Control Status pane** displays whether vSmart and vEdge devices are connected to the required number of vSmart controllers.

Each vSmart controller must connect to all other vSmart controllers in the network.

 © 2020 NterOne Corporation

Each vEdge router must connect to the configured maximum number of vSmart controllers.

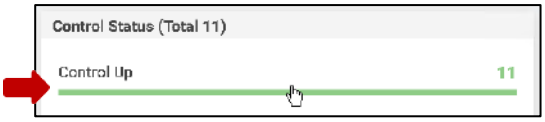


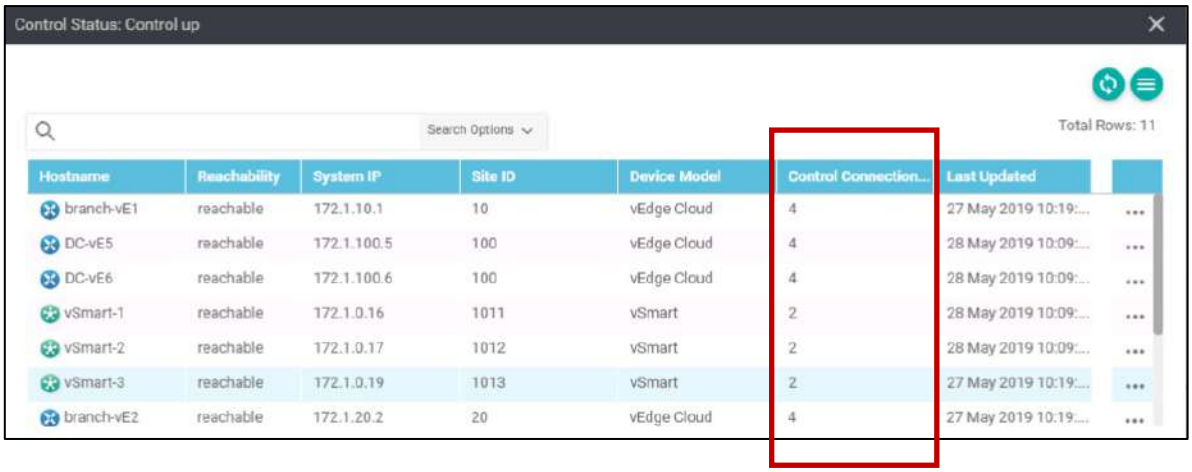
Control Up:	Total number of devices with the required number of operational control plane connections to a vSmart controller.
Partial:	Total number of devices with some, but not all, operational control plane connections to vSmart controllers.
Control Down:	Total number of devices with no control plane connection to a vSmart controller.

Step 12 **Click any row** to display a table with device details. Notice the **control connections column.**

The number in the display indicates **control per device**.

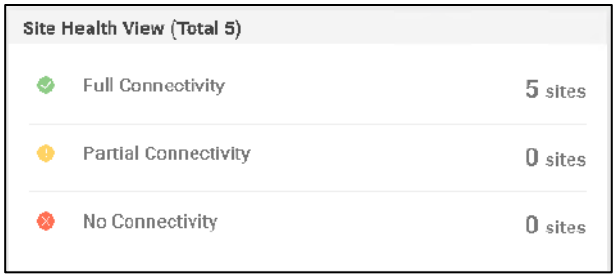
To view the **actual number of control connections** on a device click a row to view details (arrow).





Hostname	Reachability	System IP	Site ID	Device Model	Control Connection...	Last Updated
branch-vE1	reachable	172.1.10.1	10	vEdge Cloud	4	27 May 2019 10:19:...
DC-vE5	reachable	172.1.100.5	100	vEdge Cloud	4	28 May 2019 10:09:...
DC-vE6	reachable	172.1.100.6	100	vEdge Cloud	4	28 May 2019 10:09:...
vSmart-1	reachable	172.1.0.16	1011	vSmart	2	28 May 2019 10:09:...
vSmart-2	reachable	172.1.0.17	1012	vSmart	2	28 May 2019 10:09:...
vSmart-3	reachable	172.1.0.19	1013	vSmart	2	27 May 2019 10:19:...
branch-vE2	reachable	172.1.20.2	20	vEdge Cloud	4	27 May 2019 10:19:...

Step 13 **The Site Health View pane** displays the state of a site's data connections. When a site has multiple vEdge routers, this pane displays the state for the entire site, not for individual devices.



The Site Health View pane displays **three states:**

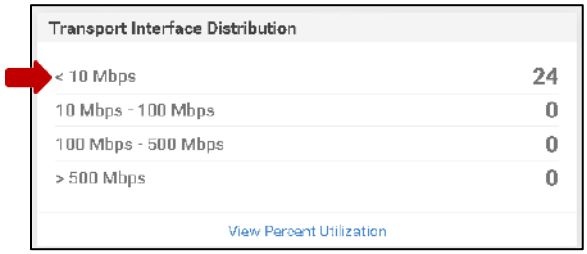
Full WAN Connectivity: Total number of sites where all BFD sessions on all vEdge routers are in the up state.

Partial WAN Connectivity: Total number of sites where a TLOC or a tunnel is in the down state. These sites still have limited data plane connectivity.

No WAN Connectivity: Total number of sites where all BFD sessions on all vEdge routers are in the down state. These sites have no data plane connectivity.

Step 14 **The Transport Interface Distribution pane** displays interface usage in the last 24 hours for all vEdge interfaces in VPN 0. This includes all TLOC interfaces.

Click a row to see details of interface usage (arrow).



Step 15 The WAN Edge Inventory pane provides four counts.

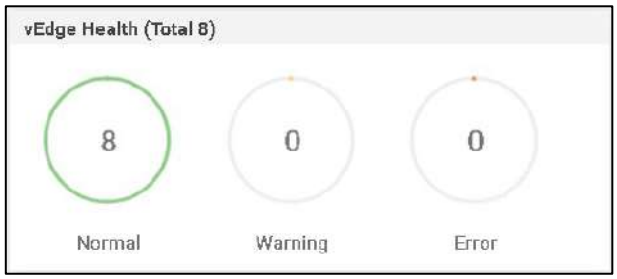
Click on Total to view the vManage vEdge inventory (arrow).

 © 2020 NterOne Corporation

Total:	Total number of vEdge routers whose authorized serial number has been uploaded on the vManage server. The serial number is uploaded in the Configuration → Devices screen.
Authorized:	Total number of authorized vEdge routers in the overlay network. These are routers marked as Valid in the Configuration → Certificates → vEdge List screen.
Deployed:	Total number of deployed vEdge routers. These are routers marked as Valid that are now operational in the network.
Staging:	Total number of vEdge routers in staging state. These are routers you configure at a staging site before shipping them to the actual branch and making them a part of the overlay network. These routers do not take part in any routing decisions nor do they affect network monitoring through the vManage NMS.

Step 16 **The vEdge Health pane** displays an aggregated view for each router state and a count of how many vEdge routers are in that state, thereby describing the health of the hardware nodes.

Click the number or the state to display a table with the last 12 or 24 hours of memory usage, CPU utilization, and hardware-related alarms, including temperature, power supply, and PIM modules.



The three states are:

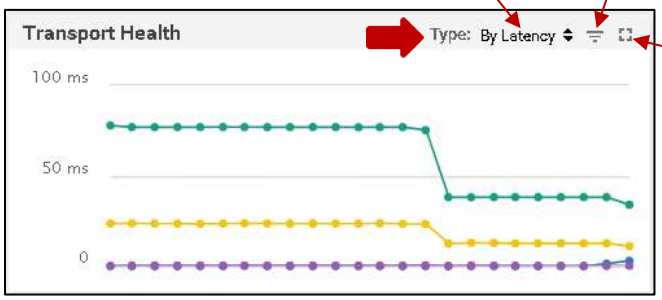
Normal:	Number of routers with memory, hardware, and CPU in normal state. Using less than 70% of total memory or total CPU is classified as normal.
Warning:	Number of routers with memory, hardware, or CPU in warning state. Using between 70% and 90% of total memory or total CPU is classified as a warning.
Error:	Number of routers with memory, hardware, or CPU in error state. Using more than 90% of total memory or total CPU is classified as an error.

Step 17 **The Transport Health pane** displays the aggregated average loss, latency, and jitter for all links and all combinations of colors.

For **Type,** select **By Latency** (arrow).

From the Type drop-down, select loss, latency, or jitter.

Click the Filter icon to select a time period for which to display data.



Click the Expand icon to open the Transport Health pop-up window. This full-screen window displays a more detailed view of the same information.

To display the information in **tabular format, click the Details tab** (arrow).

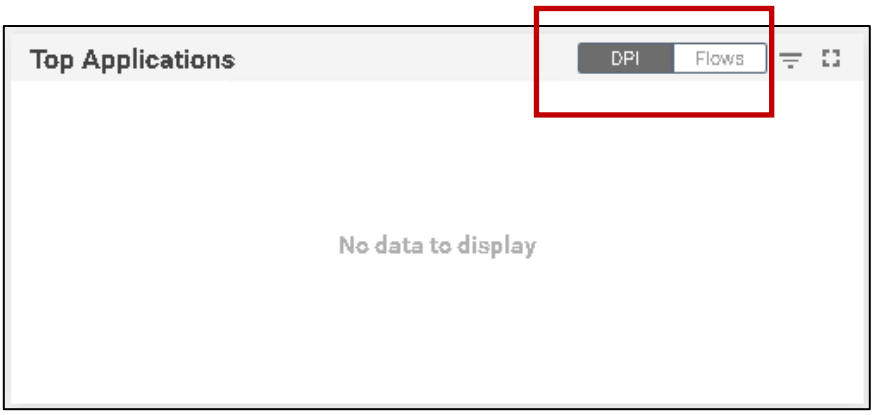
You can change the displayed type and **time period** as described above

Step 18 **The Top Applications pane** displays DPI and cflowd flow information for traffic transiting vEdge routers in the overlay network.

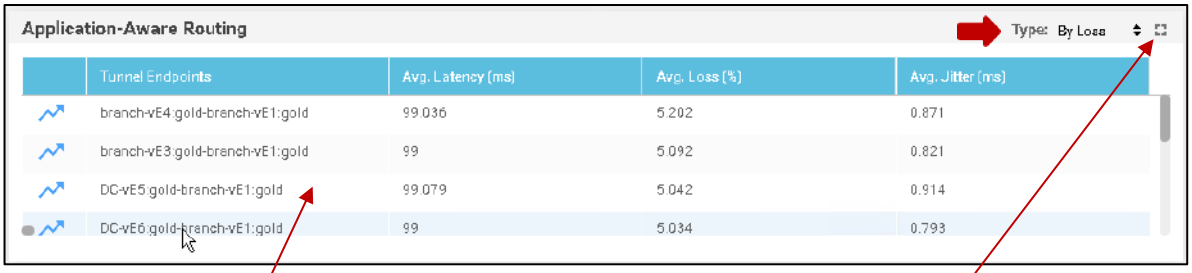
By default the pane shows **DPI flow information**.

To display cflowd flow information, click the Flows tab

 © 2020 NterOne Corporation



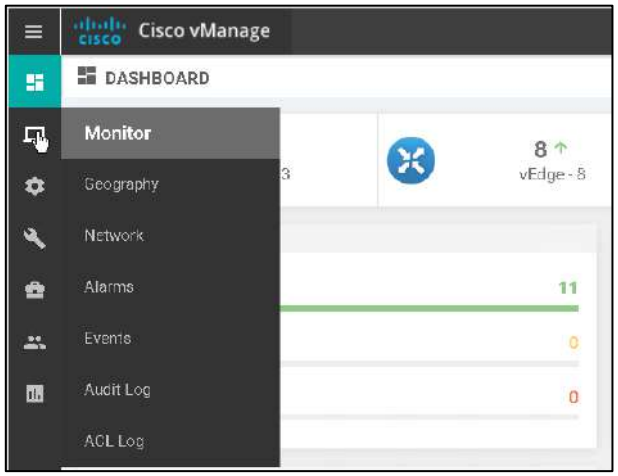
Step 19 **The Application-Aware Routing pane** displays the **10 worst tunnels based on criteria** you specify from the Type drop-down list, including loss, latency, and jitter (arrow).



Click any row to display a graphical representation of the data. Select a time period for which to display data or click Custom to display a drop-down for specifying a custom time period.

Click the Expand icon to open the Application-Aware Routing pop-up window. This full-screen window displays the 25 worst tunnels based on criteria you specify from the Type drop-down list, including loss, latency, and jitter

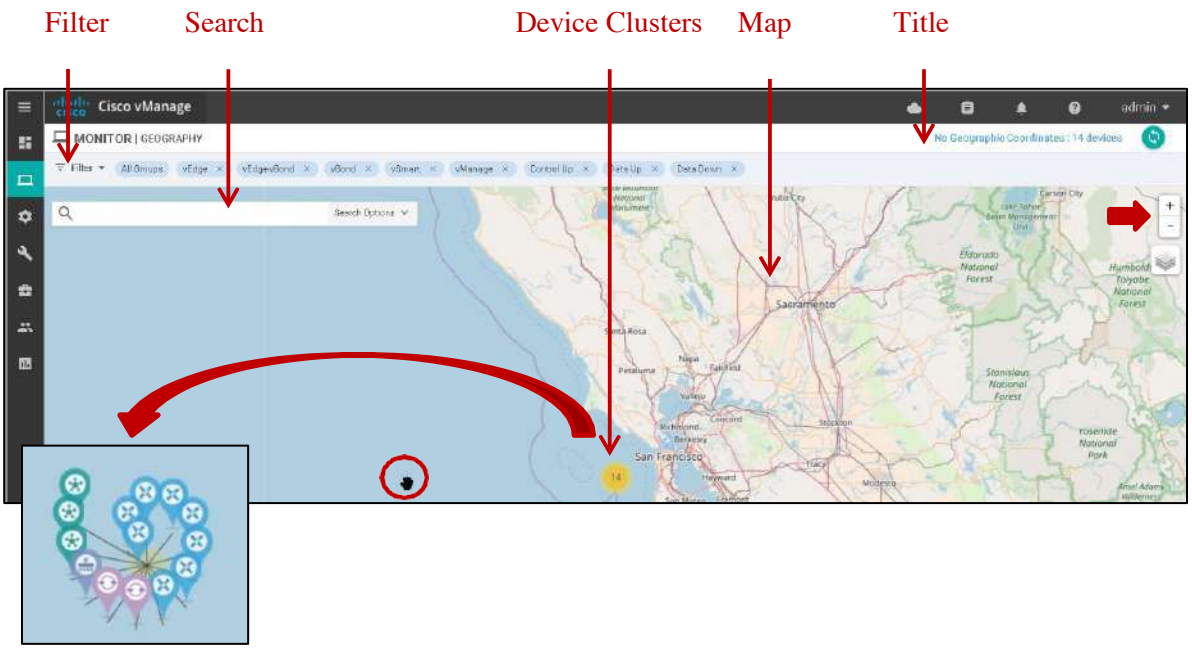
Step 20 Go to the Monitor Menu.



Step 21 Go to **Monitor > Geography**.

Zoom out a few times "-" (arrow). It may take a moment or two for the map to display properly. You can **pan around the map** with the mouse (oval).

Click the cluster to display the individual device icons on the map.



Title bar: On the right is a status message, No Geographic Coordinates, followed by the number of devices for which geographic coordinates are not configured

Filter bar: Includes the Filter button and the following tabs which reflect the default Filter selections:

- All Groups for the device group.
- vEdge, vEdge-vBond, vBond, vSmart, and vManage for the device types.
- Control Up, Data Up, and Data Down for the link states.

Map: Displays the geographic location of devices based on the latitude and longitude configured on the devices. The map has the following elements:

- **Search box:** The search applies to all devices, including those that have no geographic coordinates defined.
- + (plus) and – (minus) zoom icons.

 © 2020 NterOne Corporation

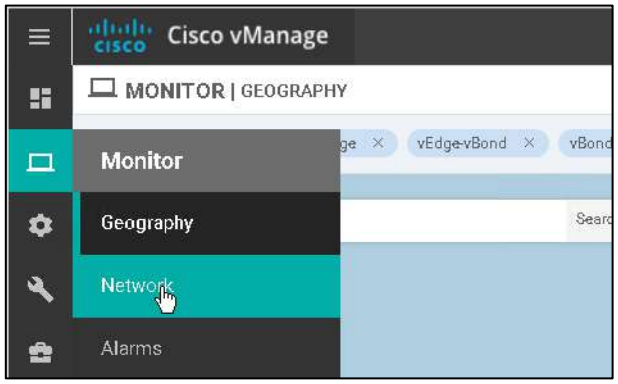
- Map provider icon, to choose any open source map provider.
- Device icons for the Cisco devices:

Device clusters: A circle with a number in the center indicates a cluster of devices located in an area.

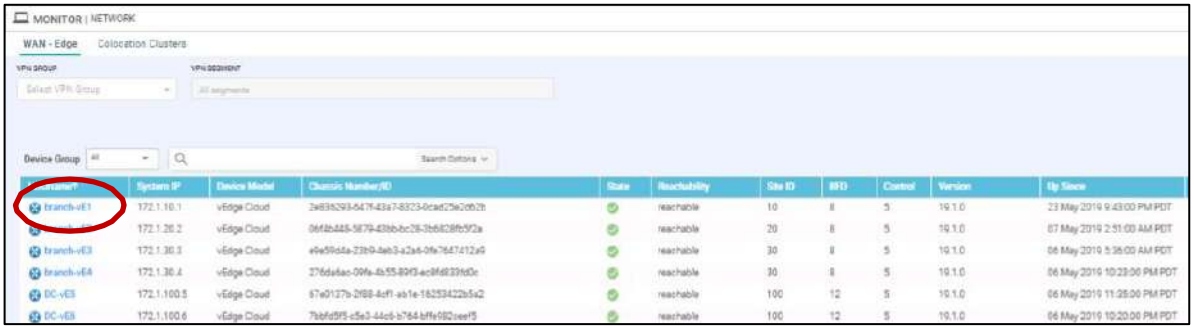
Step 22 For a device to be displayed in a specific location **on the map,** coordinates need to be configured**.** The following commands are used.

```
device(config)# system
device(config-system)# gps-location latitude degrees.minutes.seconds
device(config-system)# gps-location longitude degrees.minutes.seconds
```

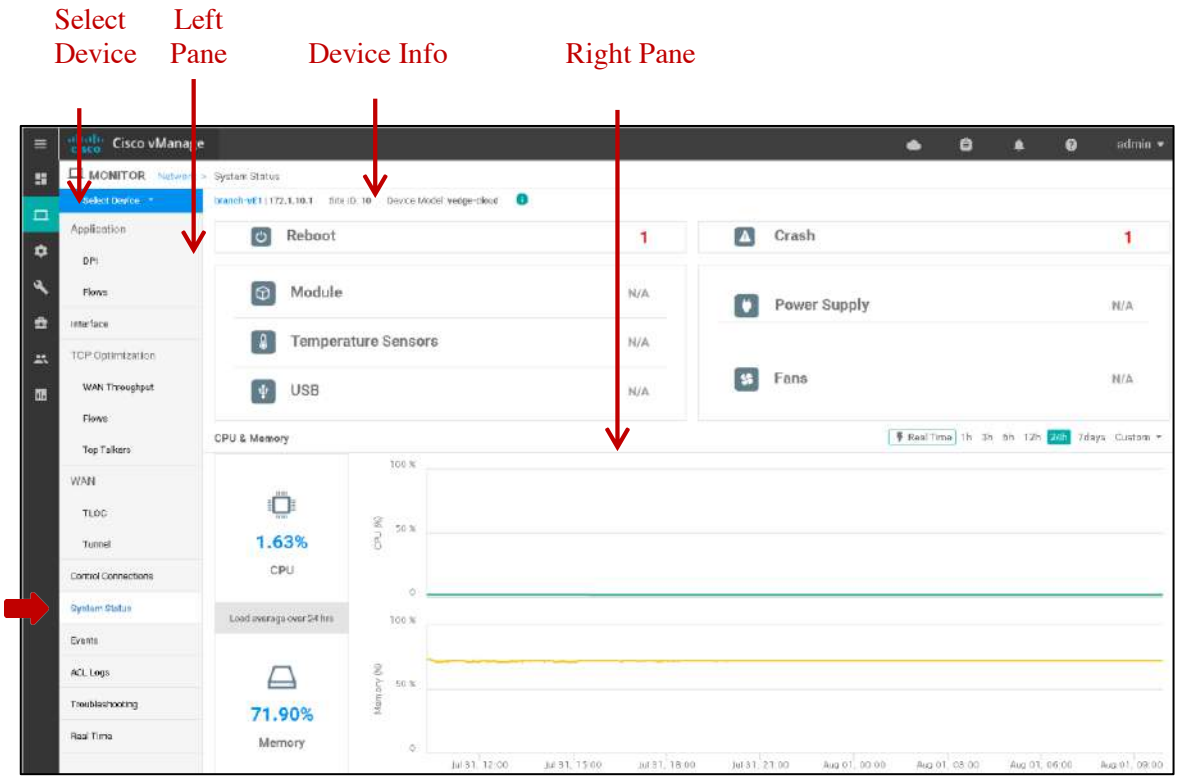
Step 23 Go to **Monitor > Network.**



Step 24 **A list of all devices** in the SD-WAN network is displayed. **Click on a device** from the list.



Step 25 After you select a device by clicking its hostname, the screen changes and displays the following options. Clicking the **System Status** link within the left-hand inventory pane will display Device Info, CPU and Memory Utilization.



Select Device: A horizontal bar that provides you:

- Select Device drop-down

Device Info: A horizontal bar that includes these elements:

- Device name
- Device IP address Device site location Device model
- More Info drop-down

Left Pane: A vertical pane that lists the categories of info you can display about the device.

- **Application**: DPI and cflowd flow information.
- **Interface**: Interface status and statistics.
- **TCP Optimization**: Statistics related to fine-tuning the processing of TCP data traffic.
- **WAN**: TLOC and tunnel status and statistics.
- **Control Connections**: Status and statistics for control connections.

 © 2020 NterOne Corporation

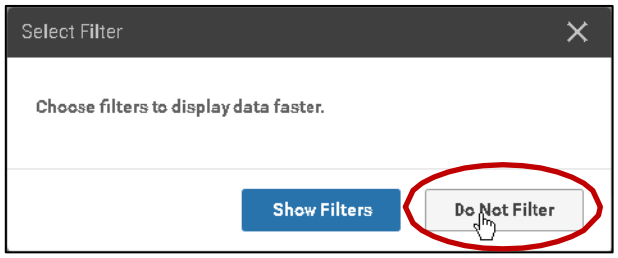
- **System Status**: Reboot and crash information, hardware component status, and CPU and memory usage.
- **Events**: Latest syslog events.
- **ACL Logs**: Logging files for access lists (ACLs).
- **Troubleshooting**: Ping and traceroute traffic connectivity tools.
- **Real Time**: Real-time device information for feature-specific operational commands.

Right Pane: Displays information about the selected category.

Step 26 Select **branch-vE1** from the Select Device pull-down menu and then click the **Real Time** link from the Left Pane.

Step 27 Select **BFD Sessions** from the **Device Options** drop-down list (arrow).

Step 28 Click on Do Not Filter.



Step 29 All the **BFD Sessions** (IPSec Tunnel keep-alive) for the device will be displayed.

System IP	Site ID	State	Source TLOC Color	Remote TLOC Color	Source IP	Destination Public IP	Destination Public Port	Encapsulation	Source Port
172.1.20.2	20	up	gold	gold	1.1.7.11	1.1.10.12	12346	ipsec	12386
172.1.30.3	30	up	gold	gold	1.1.7.11	1.1.12.13	12346	ipsec	12386
172.1.30.4	30	up	gold	gold	1.1.7.11	1.1.13.4	12346	ipsec	12386
172.1.100.5	100	up	gold	gold	1.1.7.11	1.1.15.15	12346	ipsec	12386
172.1.100.6	100	up	gold	gold	1.1.7.11	1.1.17.16	12346	ipsec	12386

Step 30 Select **Control Connections** from the Left Pane.

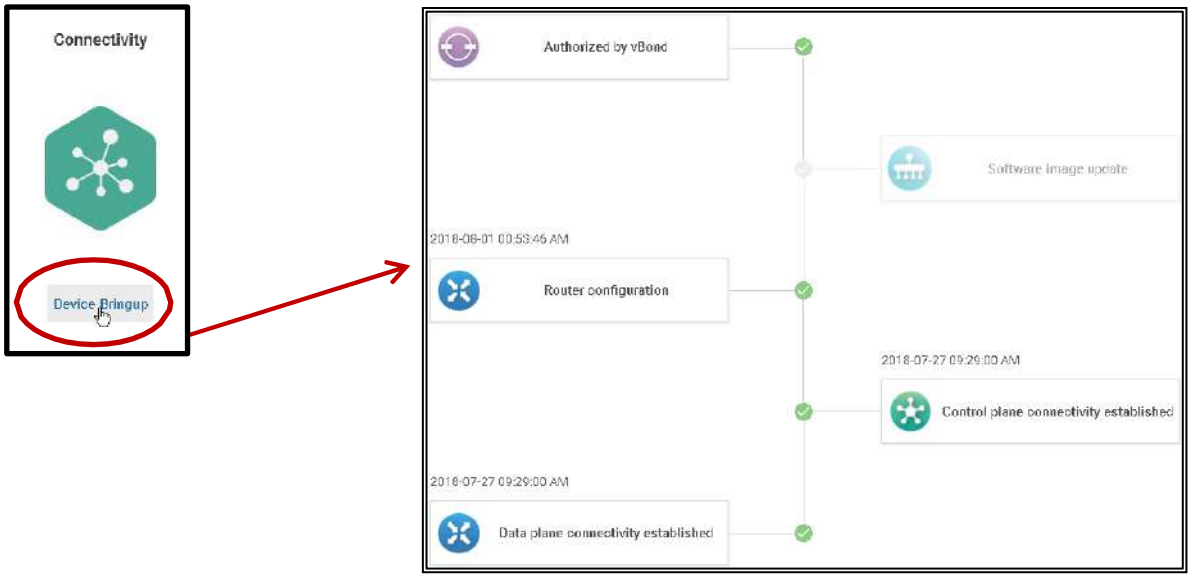
Total Control Connections to **vSmart Controllers** will be displayed. (Across all tunnels)

Step 31 Select **Troubleshooting** from the Left Pane.

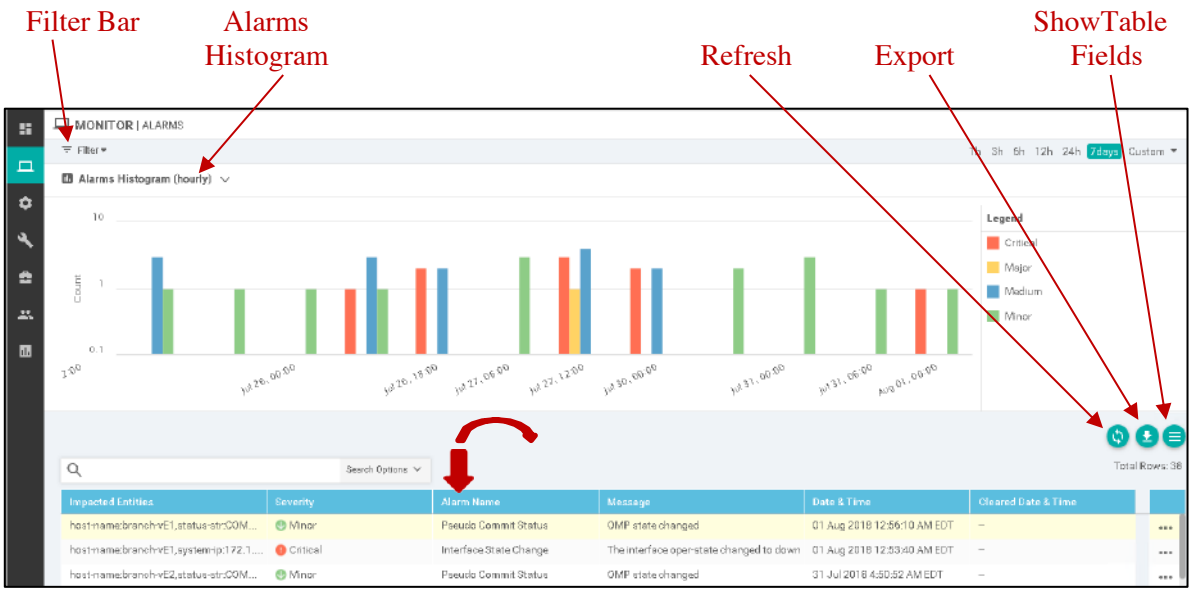
Step 32 Select **Device Bringup** from the **Connectivity** section.

Device Bringup **helps to determine if any of the device bringup phases have failed.**

 © 2020 NterOne Corporation



Step 32 Go to **Monitor > Alarms.**



Alarms Histogram: Displays a graphical representation of all alarms in order of severity: **Critical, Major, Medium, Minor.** To hide the alarms histogram, click the Alarms Histogram title or the down angle bracket to the right of it.

Filter Bar: Includes the Filter drop-down and time periods. Click the Filter icon to display a drop-down menu to add filters for searching alarms. Click a predefined or custom time period for which to display data.

Search: Includes the Search Options drop-down, for a Contains or Match string.

Refresh: Click to refresh data in the alarms table with the most current data.

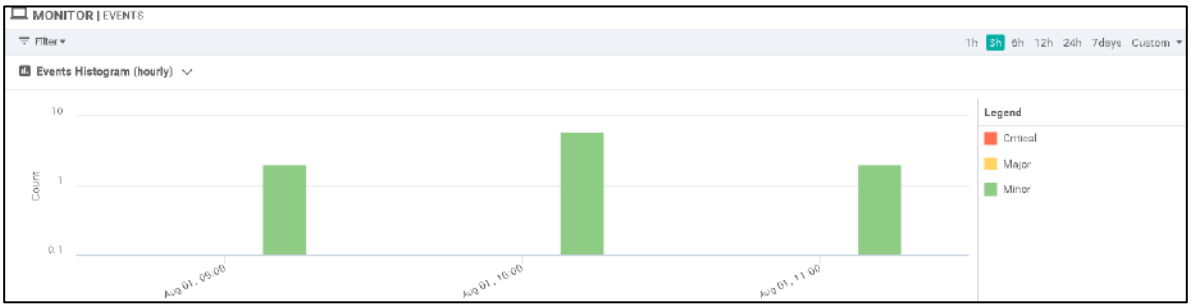
Export: Click to download all data to a file, in CSV format.

Show Table Fields: Click to display/hide columns from the table. By default, all columns are displayed.

Table of Alarms: To re-arrange the columns, drag the column title to the desired position **(arrow).**

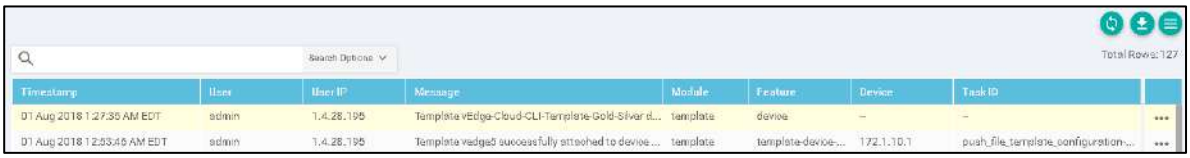
Step 33 Go to **Monitor > Events.**

Use the Events screen to display detailed information on events generated by Cisco devices



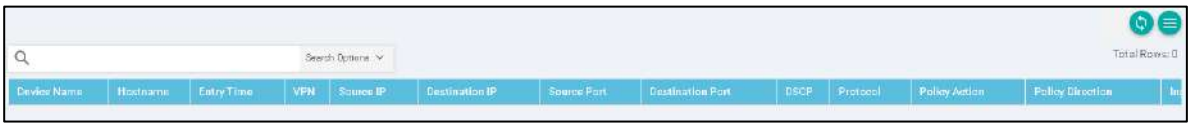
Step 34 Go to **Monitor > Audit Logs.**

Use the Audit Log screen to display a log of all activities performed on vManage.



Step 35 Go to **Monitor > ACL Log.**

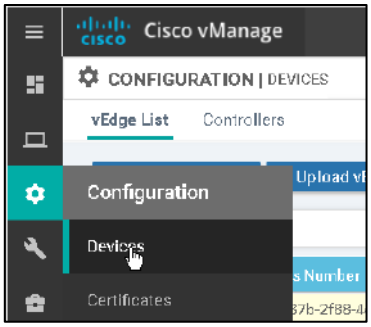
Use this screen to view logs for access lists (ACLs) configured on a vEdge router. Routers collect **ACL logs every 10 minutes**.



Step 36 Go to the **Configuration** menu.

Step 37 Go to **Configuration > Devices.**

 © 2020 NterOne Corporation



Step 38 Use the Devices screen to **add or delete devices** from the overlay network.

By default the **WAN Edge List** is displayed.

vEdge authorized serial number file contains the serial numbers of all valid vEdge routers in your network. **You receive this file from Cisco**, mark each vEdge router as valid or invalid, and then from vManage, send the file to the controller devices in the network.

Change mode: Switch between vManage and CLI mode.

Upload vEdge List: Upload the vEdge router authorized serial number file to vManage.

Export Bootstrap Configuration: Generate and download a bootstrap configuration for multiple vEdge Cloud routers (ZTP).

Step 39 Click **…** to view the **actions menu** for a device.

Step 40 The following options are available from the actions menu (arrow – previous page)

Running Configuration: View the running configuration for a device.

Local Configuration: View the device's configuration created using templates.

Delete vEdge: Deleting a vEdge router removes its serial and chassis numbers from the vEdge serial number list and permanently removes the router's configuration from vManage.

Copy Configuration: When you are replacing one router at a site with another router, you copy the old router's configuration to the new router. Then you remove the old router from the network and add the new one

Decommission vEdge: Decommissioning a vEdge Cloud router removes the device's serial number from vManage and generates a new token for the device

Change Device Values: If a Device Template is attached, and once you push the configuration to a device, you can change the value assigned to any variables.

Template Log: View a log of activities related to creation of configuration templates and the status of attaching configuration templates to devices

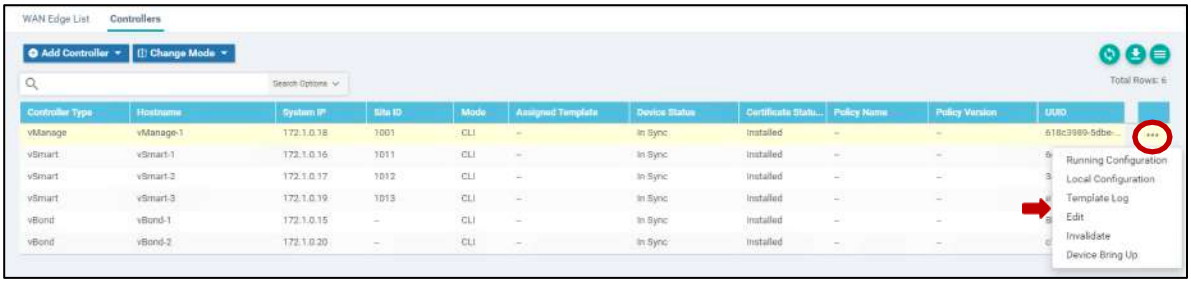
Device Bringup: Verify the status of the bringup of a vEdge router.

Step 41 Click Controllers.

Add Controller: Allows you to add a vSmart or vBond controller to the overlay network.

Change mode: Switch between vManage and CLI mode.

Step 42 Click … to view the actions menu for a controller device.



Options below were not defined previously (arrow).

Edit: Edit the IP address and login credentials of a controller device.

 © 2020 NterOne Corporation

Invalidate: This will permanently delete a controller and its connections from the network.

Step 43 **Go to Configuration > Certificates.**

Use the Certificates screen to **manage certificates and authenticate devices** in the network.

Signed certificates are used to authenticate devices in the overlay network. Once authenticated, devices can establish secure sessions between each other.

The **WAN Edge List** is displayed by default.

Send to Controllers: Send the vEdge router chassis and serial numbers to the controllers in the network (**arrow**).

Invalid: When you mark a vEdge router invalid and then push the list with **Send to Controllers**, this will **remove the router** from the **Valid vEdge List**.

Staging: When you initially bring up and configure a vEdge router, you can place it in staging state using vManage. **When the router is in this state, you can configure the router, and you can test that the router is able to establish operational connections with the vSmart controller and vManage**.

Valid: When you add vEdge routers to the network using the **Configuration → Devices** screen, you can automatically validate the routers and send their chassis and serial numbers to the controller devices by clicking the checkbox **Validate the uploaded vEdge List and send to controllers**.

If you do not select this option, you must individually validate each router and send their chassis and serial numbers to the controller devices

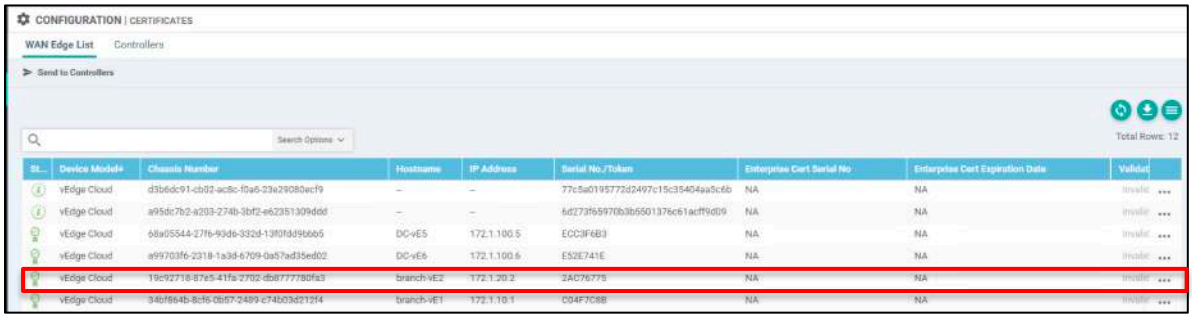
Step 44 **Remove a vEdge router by marking it <u>Invalid</u>**. Perform the following steps.

Step 45 First, view the **Valid vEdge List** on vBond-1**.**

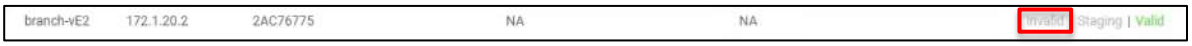
Login to vBond-1 (SSH – 10.82.83.132) and use the following command. Notice the highlighted entry (branch-vE2).

```
vBond-1# show orchestrator valid-vedges

 CHASSIS NUMBER                          SERIAL NUMBER                    VALIDITY      ORG
-----------------------------------------------------------------------------------------------
------
19C92718-87E5-41FA-2702-DB8777780FA3     2AC76775                         valid  NterOne Lab
34BF864B-8CF6-0B57-2489-C74B03D212F4     C04F7C8B                         valid  NterOne Lab
68A05544-27F6-93D6-332D-13F0FDD9BBB5     ECC3F6B3                         valid  NterOne Lab
A95DC7B2-A203-274B-3BF2-E62351309DDD     6d273f65970b3b5501376c61acff9d09 valid  NterOne Lab
A99703F6-2318-1A3D-6709-0A57AD35ED02     E52E741E                         valid  NterOne Lab
CSR-06D5C3D2-822A-52B8-6206-B49F7F1DEB07 A7AD48E5                         valid  NterOne Lab
CSR-10A99631-C803-3AD3-B9C2-8182260392C0  ef7db55209665d52262f7be9446f7a78 valid  NterOne Lab
CSR-7406BBDC-DDD1-6FDE-7734-3276A33086FC  BC4C68DF                         valid  NterOne Lab
CSR-8E1294E5-D14D-0756-185F-F344A5062D1F  e722f6f82c30d8d45bf3d513a9c5ded1  valid NterOne Lab
CSR-CF7D2D66-0435-8096-601A-E12D90AF5DAD  7DF5CF2F                         valid  NterOne Lab
CSR-EDD9529B-64DA-7D24-6EF3-56126CEF968C  81D3A1A7                         valid  NterOne Lab
D3B6DC91-CB02-AC8C-F0A6-23E29080ECF9     77c5a0195772d2497c15c35404aa5c6b valid  NterOne Lab
```

Step 46 Go back to **Configuration > Certificates** and identify the entry that was **highlighted.** Select **branch-vE2** from the list.

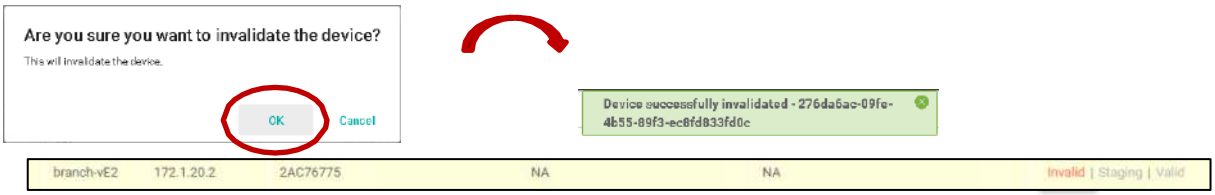


Step 47 Click **Invalid** for the branch-vE2 entry.



Step 49 Click **OK** to invalidate the device. You'll notice a message at the top of the window.

The device will be marked as **Invalid** (Red).



Step 48 The **Send to Controllers button** will be highlighted in Red.

 © 2020 NterOne Corporation

Step 49 Click Send to **Controllers**.

The vEdge List (Chassis and Serial number) will be pushed to all of the controllers in the network. You'll be **redirected to TASK VIEW** to monitor the progress.

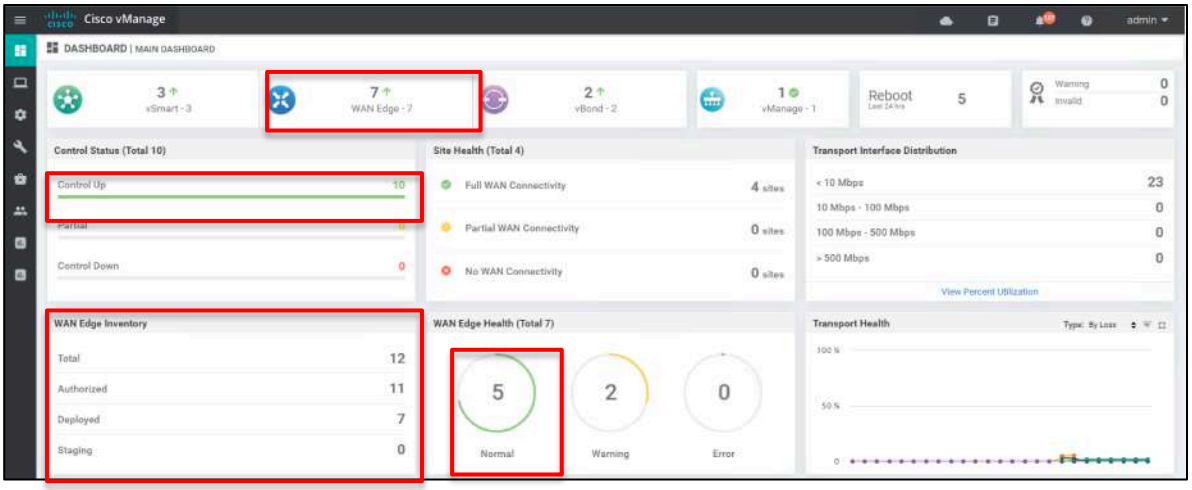
Step 50 **Go back to vBond-1** and view the **Valid vEdge List** once again.

Notice that the entry belonging to branch-vE4 is no longer listed.

```
vBond-1# show orchestrator valid-vedges
```

```
CHASSIS NUMBER                              SERIAL NUMBER                     VALIDITY  ORG
--------------------------------------------------------------------------------------------------
34BF864B-8CF6-0B57-2489-C74B03D212F4        C04F7C8B                          valid     NterOne Lab
68A05544-27F6-93D6-332D-13F0FDD9BBB5        ECC3F6B3                          valid     NterOne Lab
A95DC7B2-A203-274B-3BF2-E62351309DDD        6d273f65970b3b5501376c61acff9d09  valid     NterOne Lab
A99703F6-2318-1A3D-6709-0A57AD35ED02        E52E741E                          valid     NterOne Lab
CSR-06D5C3D2-822A-52B8-6206-B49F7F1DEB07    A7AD48E5                          valid     NterOne Lab
CSR-10A99631-C803-3AD3-B9C2-8182260392C0    ef7db55209665d52262f7be9446f7a78  valid     NterOne Lab
CSR-7406BBDC-DDD1-6FDE-7734-3276A33086FC    BC4C68DF                          valid     NterOne Lab
CSR-8E1294E5-D14D-0756-185F-F344A5062D1F    e722f6f82c30d8d45bf3d513a9c5ded1  valid     NterOne Lab
CSR-CF7D2D66-0435-8096-601A-E12D90AF5DAD    7DF5CF2F                          valid     NterOne Lab
CSR-EDD9529B-64DA-7D24-6EF3-56126CEF968C    81D3A1A7                          valid     NterOne Lab
D3B6DC91-CB02-AC8C-F0A6-23E29080ECF9        77c5a0195772d2497c15c35404aa5c6b  valid     NterOne Lab
```

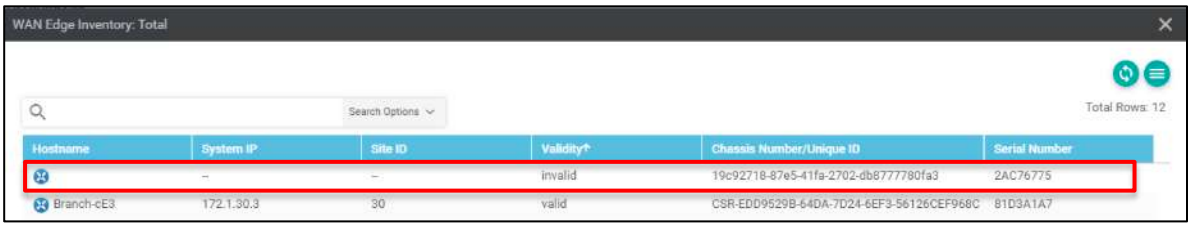
Step 51 **Go to the Dashboard** to view the results of the device being invalidated.



Step 52 **Notice that the** vEdge has been removed from most of the areas of the Dashboard <u>except for one</u>**.**

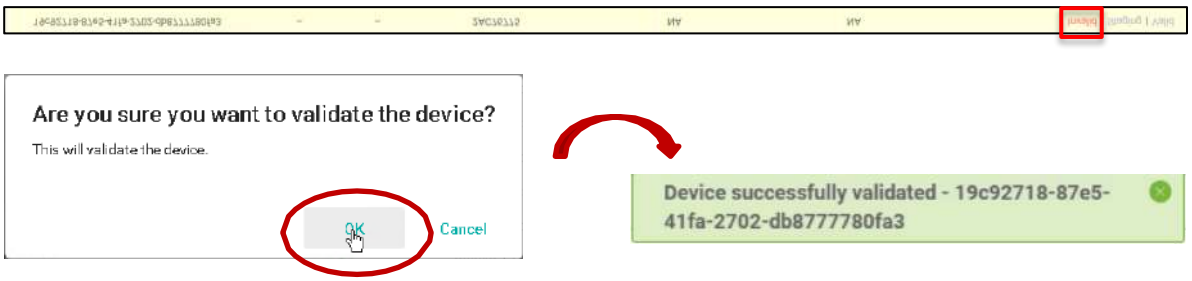
Step 53 Click **Total** from **vEdge Inventory.**

Step 54 When the window appears, **click on the Validity column** (arrow) to re-order based on validity.



Step 55 **Make sure to remark branch-vE2 <u>valid once again</u>.**

Step 56 Go to **Configuration > Certificates** and perform the following. Locate the **Invalid device**.



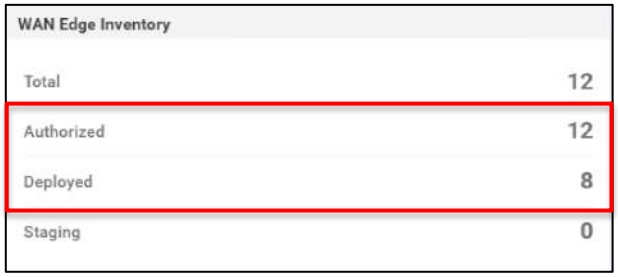
 © 2020 NterOne Corporation

Step 57 **Send to Controllers.**

Step 58 **Go to vBond-1** and verify.

```
vBond-1# show orchestrator valid-vedges

CHASSIS NUMBER                          SERIAL NUMBER       VALIDITY  ORG
----------------------------------------------------------------------------------------------
19C92718-87E5-41FA-2702-DB8777780FA3    2AC76775            valid     NterOne Lab
34BF864B-8CF6-0B57-2489-C74B03D212F4    C04F7C8B            valid     NterOne Lab
68A05544-27F6-93D6-332D-13F0FDD9BBB5    ECC3F6B3            valid     NterOne Lab
```

Step 59 **Go back to the Dashboard** and confirm your results.

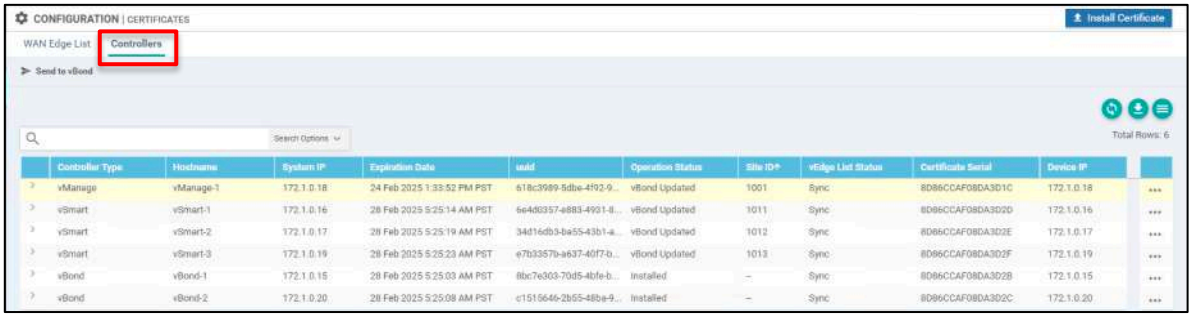


Step 60 Go to **Configuration > Certificates, and then click on Controllers.**

Signed certificates are used to authenticate devices in the overlay network. Once authenticated, devices can establish secure sessions between each other.

It is from vManage that you generate these **certificates and install them on the controller devices**.

To determine which controllers in the network are valid, the vBond orchestrator keeps a list of the controller serial numbers. vManage learns these serial numbers during the certificate-generation process.



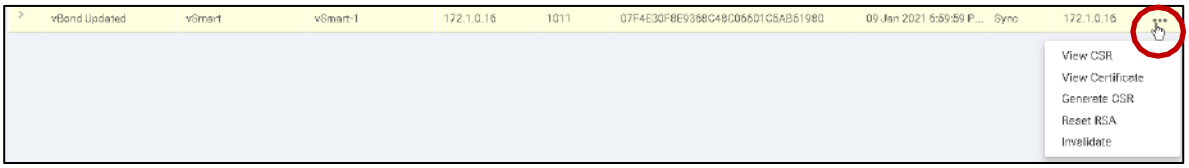
To send the controller serial numbers to the vBond orchestrator use **Send to vBond.**

Step 61 Certificates can be installed on a controller by selecting it from the list, and then use **Install Certificate**.

Step 62 Select **vSmart-1** from the list of controllers.

Step 63 Notice at the **bottom of the window**, the indication of what has been performed for this controller.

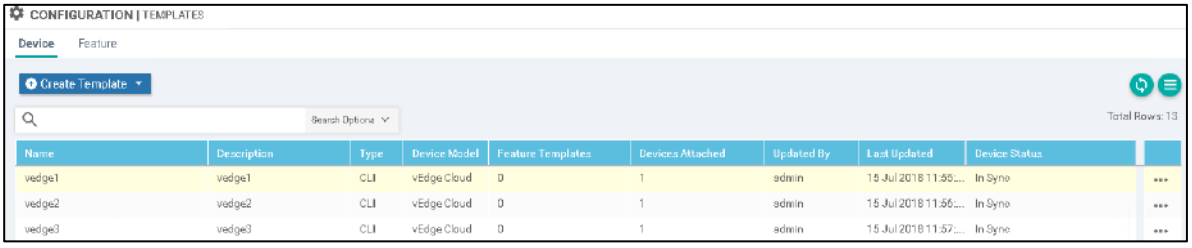
Step 64 Click on **…** to the right of the vSmart entry, to view the options from the **actions menu**.



View CSR:	View the Certificate Signing Request generated for the device.
View Certificate:	View the Installed Certificate on the device.
View CSR:	Generate a Certificate Signing Request for the device.
Reset RSA:	Reset the RSA keys used in Certificate Generation.
Invalidate:	This will permanently delete the controller and its connections from the network.

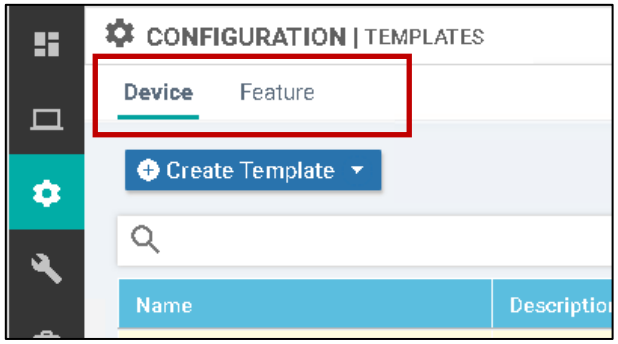
Step 65 Go to **Configuration > Templates.**

Templates define a **device's complete operational configuration**.



Step 66 The following **options are available** from the **top-left corner** of the templates window.

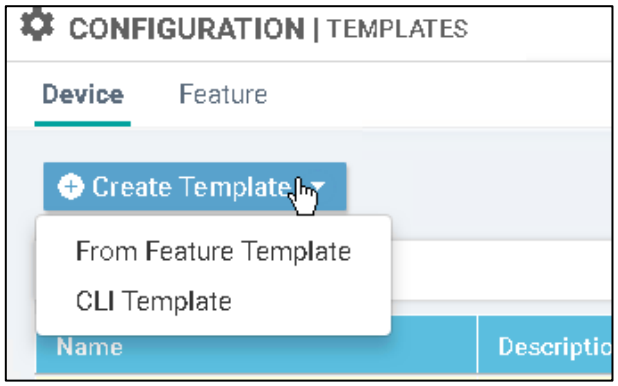
 © 2020 NterOne Corporation



Device: Device templates define a device's complete operational configuration. A device template **consists of a number of feature templates**.

Feature: Each feature template defines the configuration for a **particular software feature**. Some feature templates are mandatory, indicated with an asterisk (*), and some are optional.

Step 67 **Click + Create Template.**



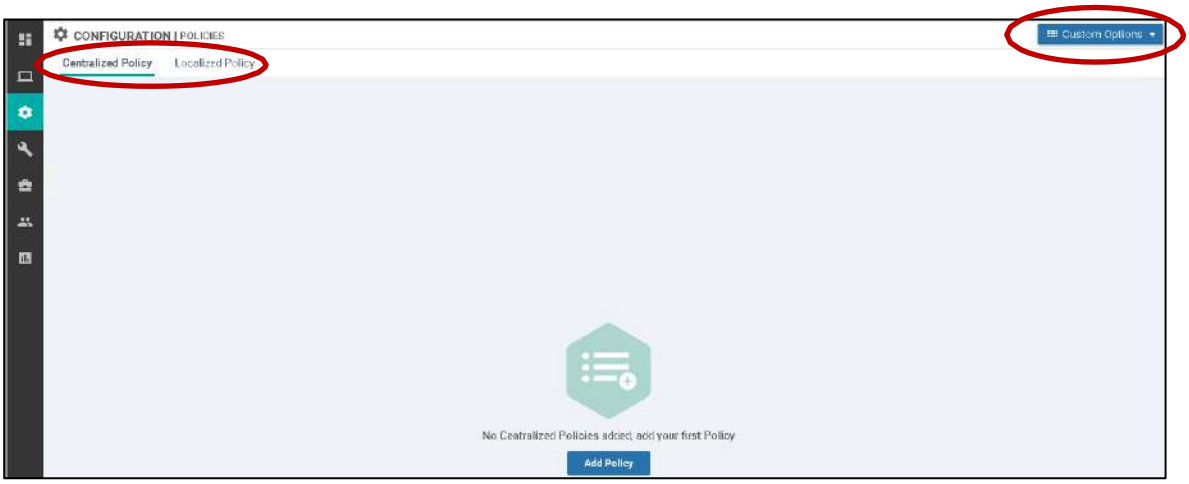
From Feature Template: Creates a Device Template using a variety of Feature Templates.

CLI Template: Create a Device Template using direct Commands.

Step 68 Templates will be covered in more detail later in the Lab Guide.

Step 69 Go to **Configuration > Policies**.

Use the Policies screen to create and activate **centralized and localized control and data policies** for vSmart controllers and vEdge routers.



Centralized Policy: Clicking Add Policy opens the policy configuration wizard. **This wizard is a UI policy builder that consists of four screens** that you use to configure centralized policy components and to modify existing components in centralized policies.

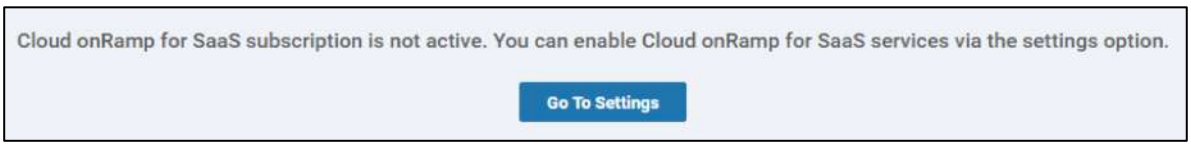
Localized Policy: In the Add CLI Policy screen, enter the **policy configuration commands**. Use the Create Variable to create a variable to use in the configuration. Use Select a File to import a file containing policy configuration commands.

Custom Options: For centralized policy, **you can create individual policy components directly ra**ther than using the policy configuration wizard.

Step 70 **Policies will be covered in greater detail later** in the Lab Guide.

Step 71 Go to **Configuration > Cloud OnRamp for SaaS.**

Cloud OnRamp for SaaS service calculates a value called the **Viptela Quality of Experience** (vQoE). The **vQoE** value weighs loss and latency using a formula customized for each application. CloudExpress service computes vQoE values for applications and paths, and then **assigns applications to the paths that best match their vQoE value.**



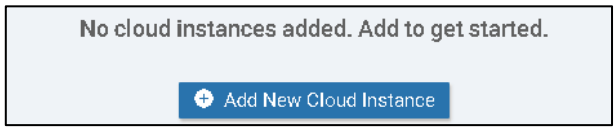
Step 72 Go to **Configuration > Cloud onramp for IaaS**.

Use the Cloud OnRamp for IaaS screen to create virtual private cloud (VPC) instances for **hosting vEdge Cloud routers in different AWS regions on the internet.**

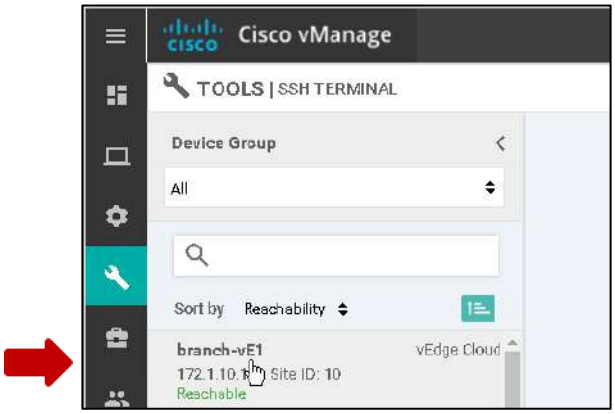
A Cloud OnRamp VPC comprises three components: a gateway VPC, which connects the overlay network to one or more cloud-based applications, a host VPC, which is where cloud-based applications reside,

 © 2020 NterOne Corporation

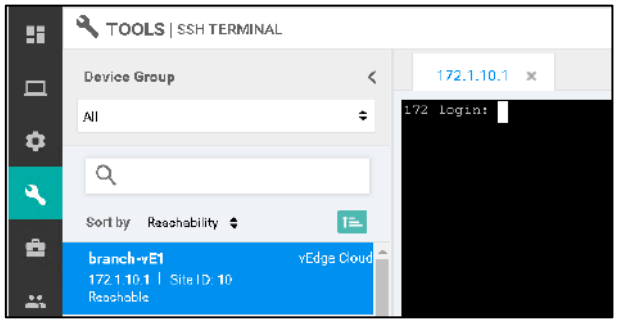
and the connections, or mappings, between the gateway VPN and one or more host VPCs



Step 73 Go to **Tools > SSH Terminal**, and then select **branch-vE1** from the list of devices.



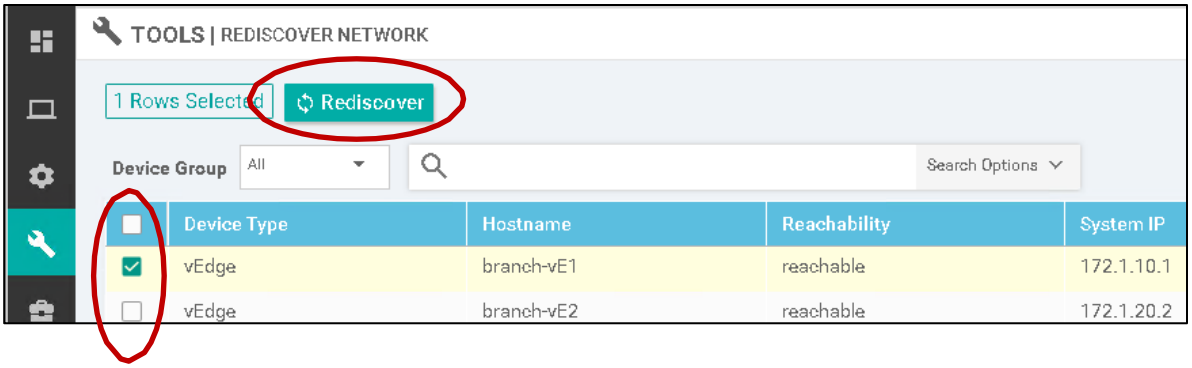
Step 74 This will **open a SSH Terminal** to the vEdge router. The Terminal area can be used for all devices that are part of the vManage inventory.



Step 75 Go to **Tools > Rediscover Network.**

This will synchronize the data on a specific device, or multiple devices with vManage.

This can be useful if there were any changes made to a device or if you want to make sure that all the properties are current.



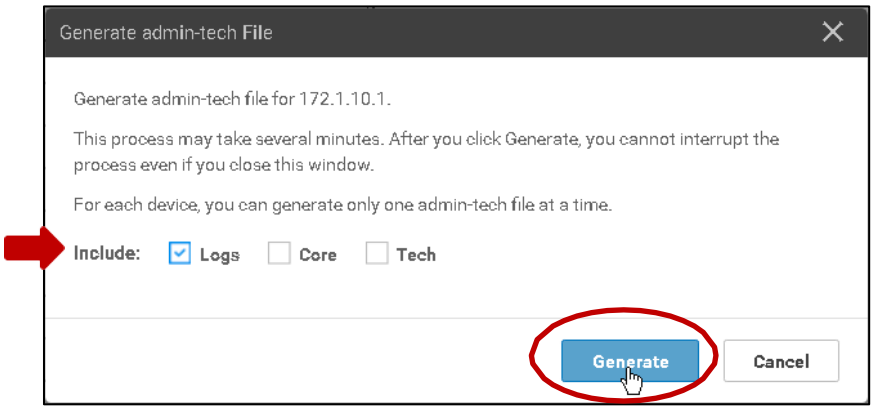
Step 76 Go to **Tools > Operational Commands.**

Select **branch-vE1** from the list, and then click **…** to view the options from the actions menu.

Step 77 Select **Admin Tech** from the menu.

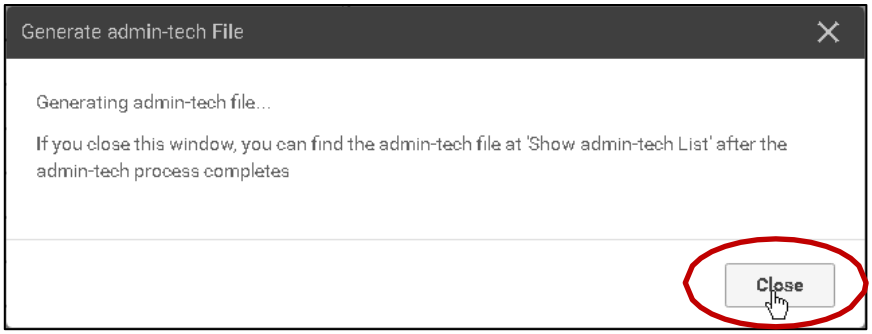
Use this to collect system status info for a device in a tar file, **to aid in troubleshooting and diagnostics**.

Step 78 Make sure that only **Logs** is selected (default), and then click **Generate**.



Step 79 The vManage will now **generate the Admin Tech file**. Click **Close** when done.

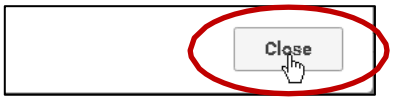
 © 2020 NterOne Corporation



Step 80 Click **Show Admin Tech List** located in the upper right-hand corner of the Tools > Operational Commands window in order to download the previously created Admin Tech File.

Step 81 Click the **Download icon** to download the file onto the computer.

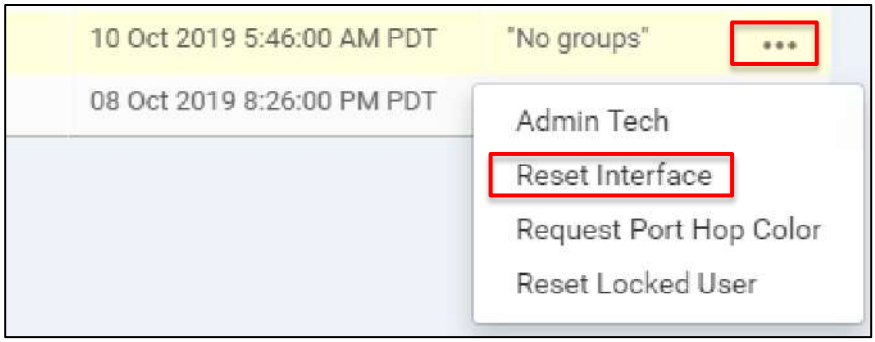
Step 82 Click **Close** when done.



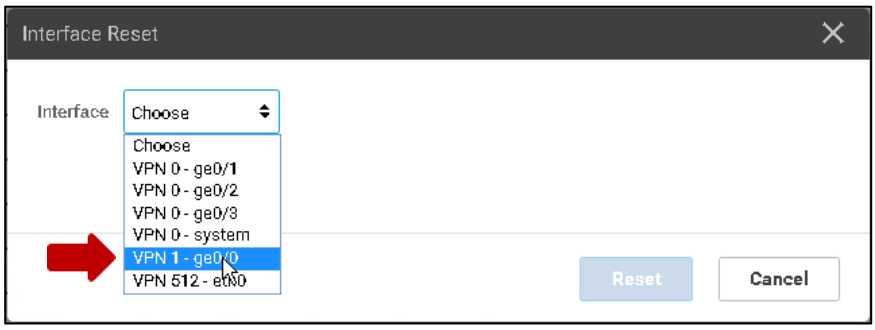
Step 83 With **branch-vE1** still selected on the list, click **Reset Interface** from the actions menu (arrow).

Use the Interface Reset command to shut down, and then restart an interface on a device in a single operation, **without having to modify the device's configuration.**

This is useful when a template is attached to the device and the CLI is locked.



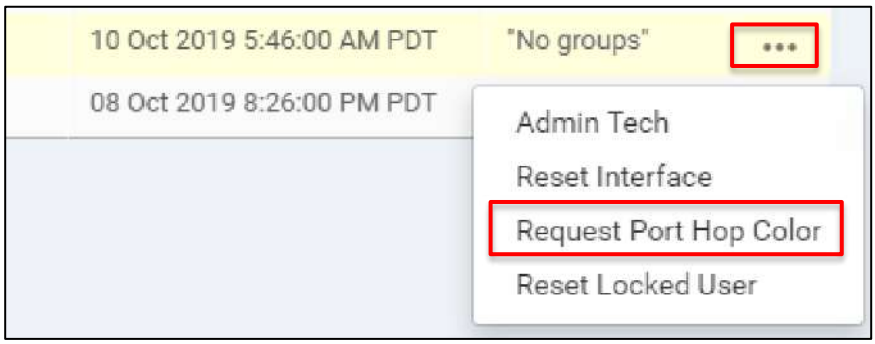
Step 84 Select **VPN 1 – ge0/0** from the **Interface** drop-down menu.



Step 85 Click **Reset**. This will reset the selected interface.

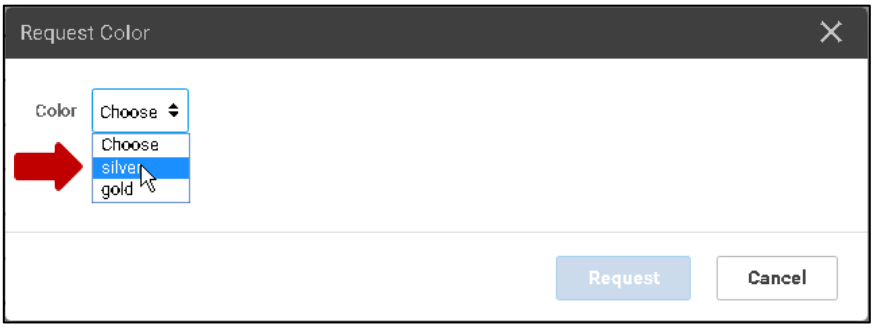
Step 86 Click the actions menu again **…** and select **Request Port Hop Color**.

This will allow you to **change the port number** used for connections from a **specific color**.



 © 2020 NterOne Corporation

Step 87 Select **silver** from the **Color** drop-down menu.

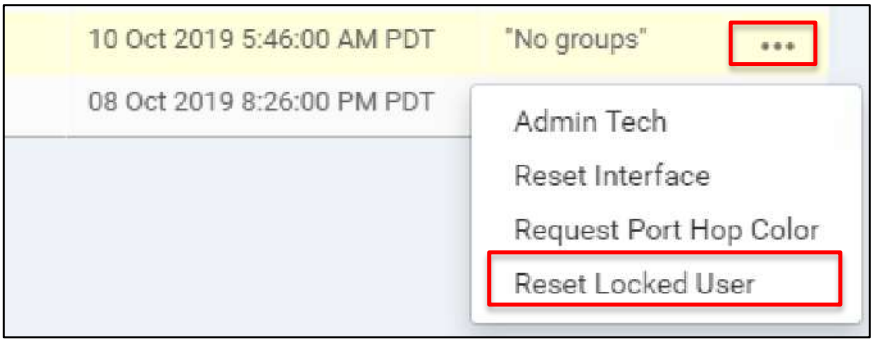


Step 88 Click **Request** when done.

Step 89 If you **port hop too frequently** you will get an **error message** saying to wait (20 minute timer).

Step 90 Click the **…** once again and select **Reset Locked User**.

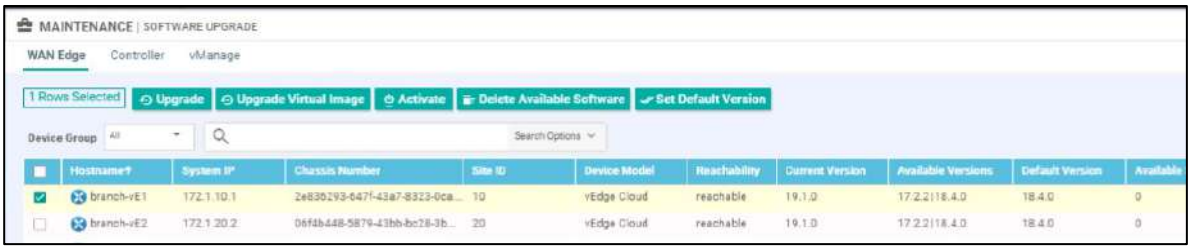
This will reset the account of a user whose account is locked. An account becomes locked when the user can no longer log in to a device.



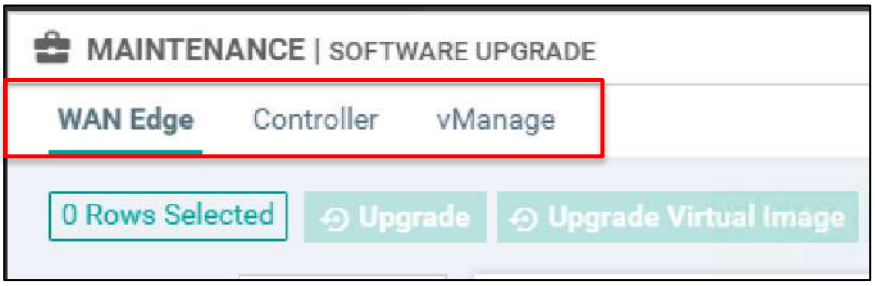
Step 91 Select **admin** from the Users drop-down menu, and then click **Reset**.

Step 92 Go to the Maintenance menu.

Step 93 Go to **Maintenance > Software Upgrade** (**Do NOT actually perform a software upgrade in the lab.**)



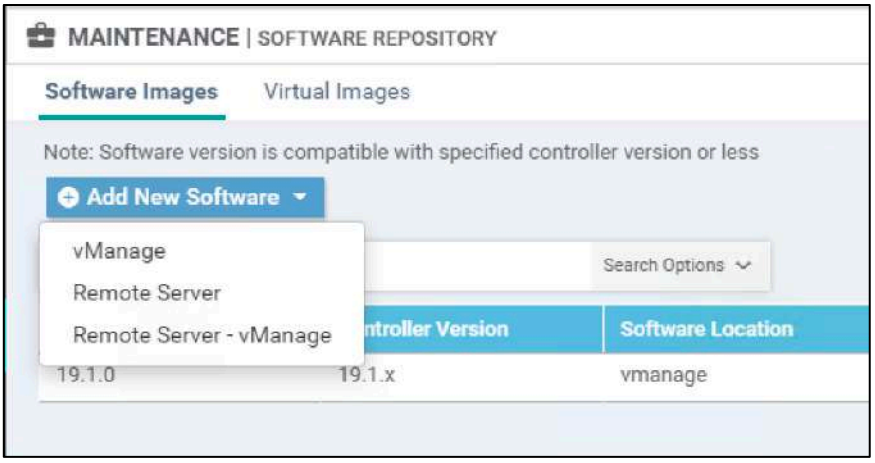
Step 94 From here you can **Upgrade any device(s)** within the vManage inventory.



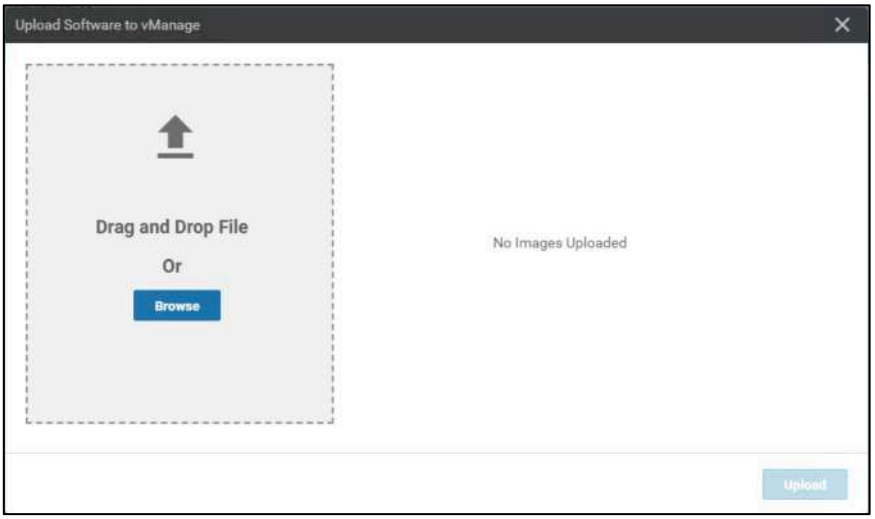
Step 95 Go to **Maintenance > Software Repository**.

Click **+ Add New Software**. Images can be added to the repository in one of 3 ways (arrow).

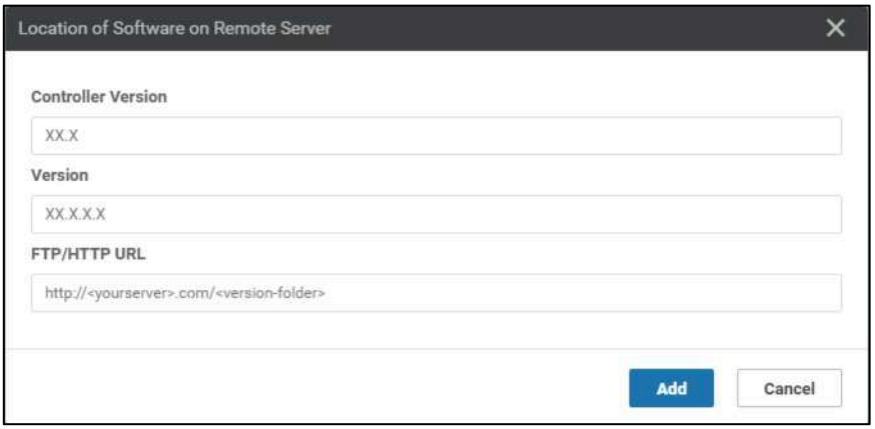
 © 2020 NterOne Corporation



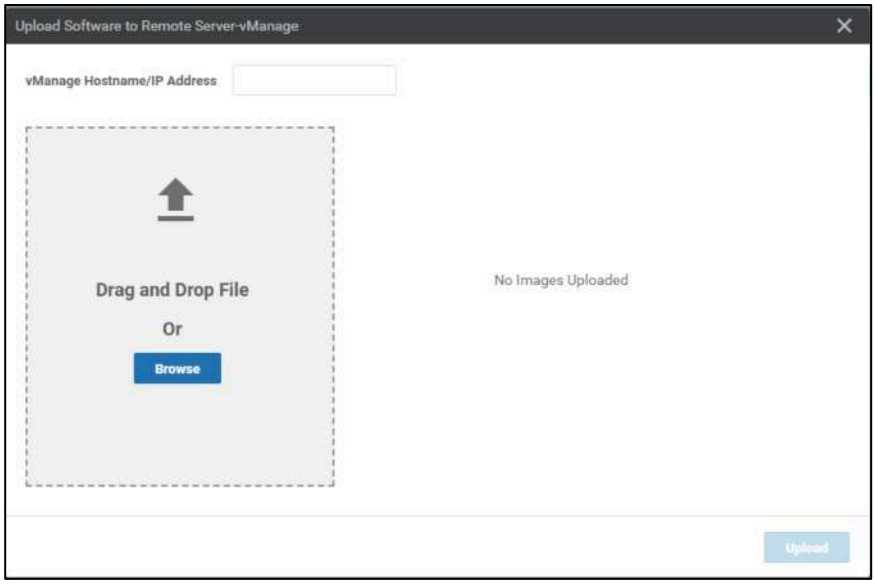
Step 96 If you select **vManage**, then the image will be stored on the **current vManage node**.



Step 97 If you select **Remote Server**, then the image will be stored on a **Remote Server such as FTP**.

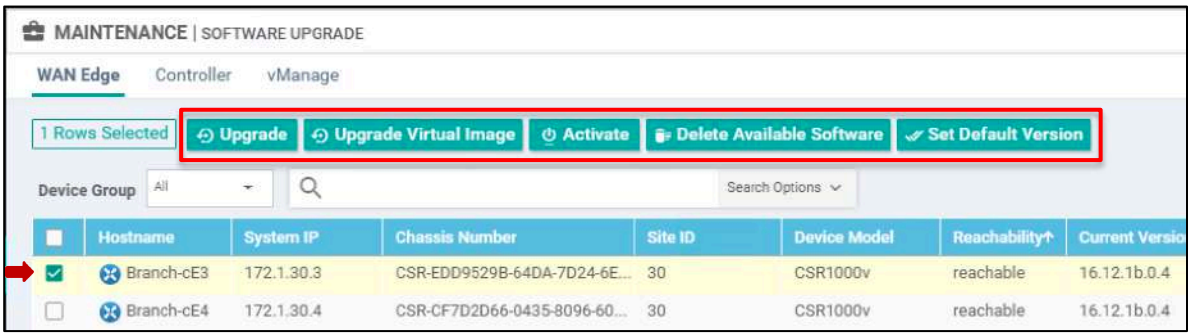


Step 98 If you select **Remote Server-vManage**, then the image will be stored on a **Remote vManage node**.



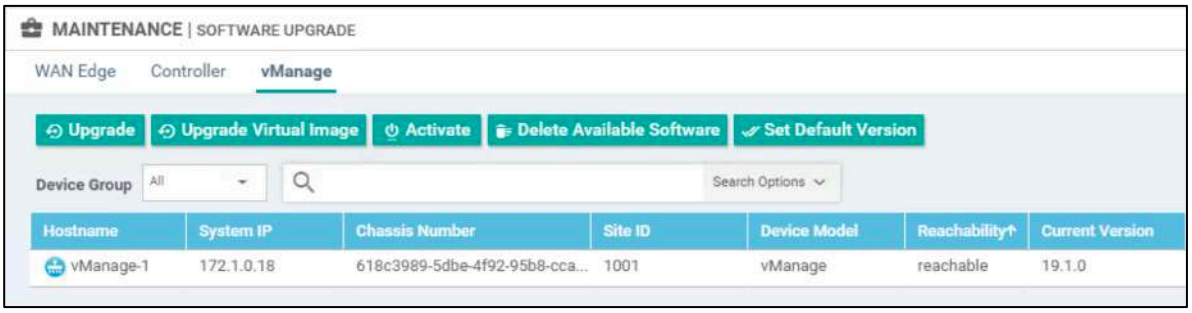
Step 99 Navigate back to **Maintenance > Software Upgrade.**

Step 100 Select the **first device** from the list. This will activate the **menu options** above it.



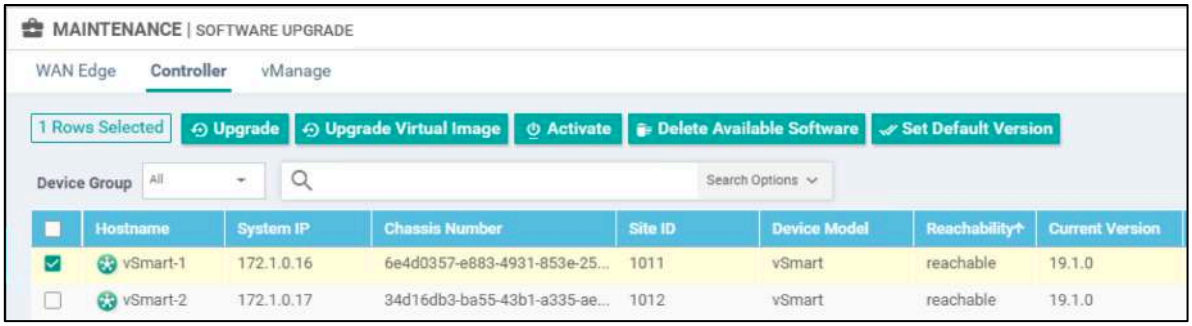
Step 101 Devices should be upgraded in the following order.

Step 102 The **vManage are upgraded first.**

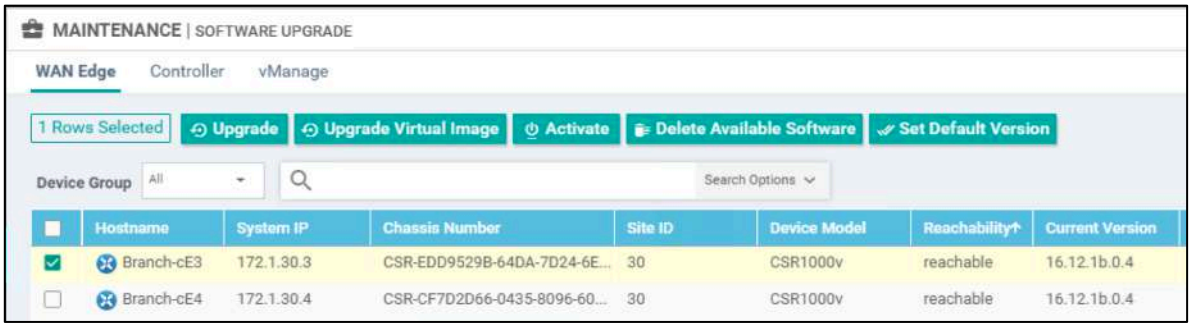


Step 103 The Controllers, vBond and vSmart are upgraded next.

 © 2020 NterOne Corporation



Step 104 The **WAN Edge routers are upgraded last.**



Step 105 The **steps to upgrade** a device are as follows:

- 1 – Upgrade the Software Image.

- 2 – Activate the Software Image.

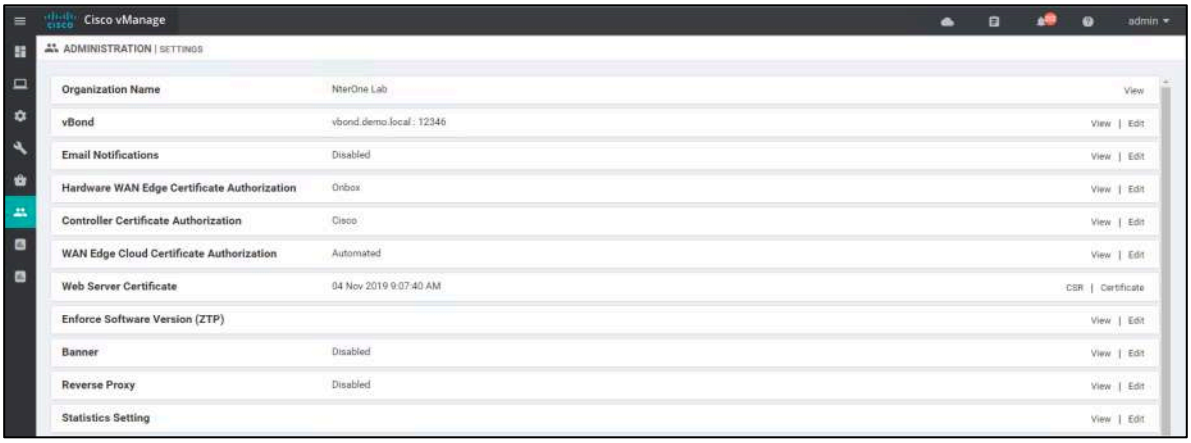
- **3 – Set the Default Software Version.**

- **Delete a Software Image (Optional).**

Step 106 Go to **Maintenance > Device Reboot.** Use the Device Reboot screen to reboot one or more devices.

Step 107 Go to the **Administration** Menu.

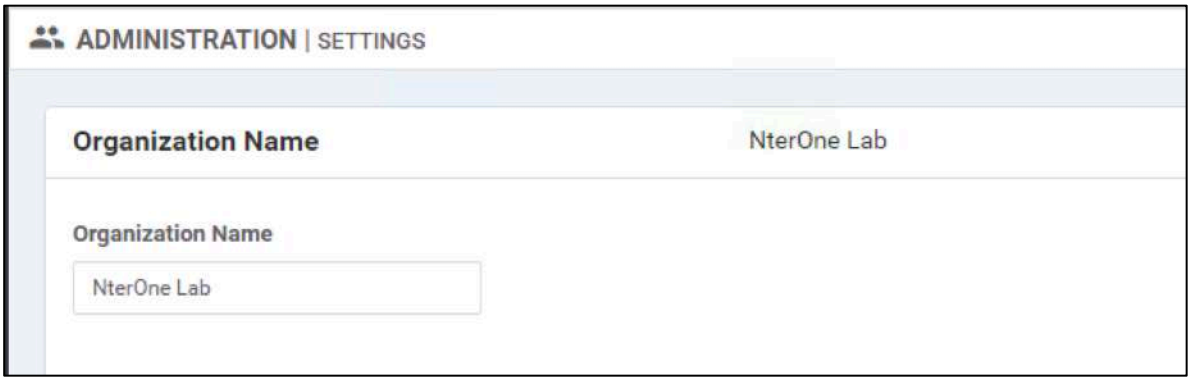
Step 108 Go to **Administration > Settings.**



Step 109 Click **View for Organization Name.**

Before you can generate a CSR, **you must configure the name of your organization**. The organization name is included in the CSR.

Once the **control connections are up and running**, the **organization name is not editable**.



Step 110 Click **View** for **vBond**.

A DNS name is used that points to the vBond orchestrator or the IP address of the vBond orchestrator and the port number to use to connect to it.

Entering a value here (preferably DNS name) will provide a vBond entry for devices using a feature based device template.

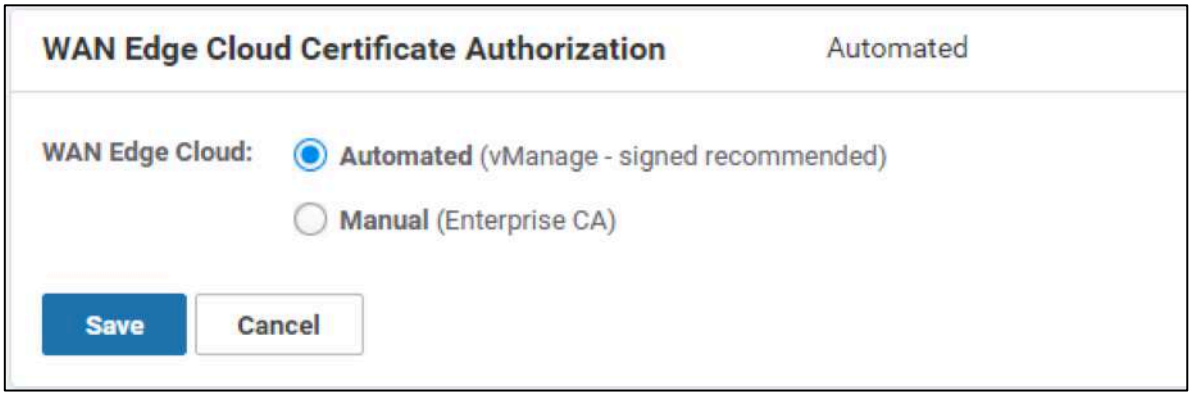
 © 2020 NterOne Corporation

Step 111 Click Edit for Controller Certificate Authorization.

You can generate CSRs and Install Certificates from vManage for all controllers. You can either approach this **manually** (default) or **automate** the process using a Symantec account.

Step 112 **Click** View **for** WAN Edge Cloud Certificate Authorization**.**

The vManage will **automatically install vEdge Cloud certificates** when onboarded (default). If using an Enterprise CA, then this process can be performed manually.



Step 113 Click Certificate for Web Server Certificate.

The vManage uses a self-signed certificate when connecting via browser. **Alternatively, a trusted certificate can be installed (Import)** for the Web Server instead.

Step 114 Click **View for Enforce Software Version (ZTP).**

This setting enforces a specific software version on a vEdge router when onboarded using ZTP.

When enabled, any vEdge router joining the network is configured with the version of the software specified in the Enforce Software Version field regardless of whether the router is running a higher or lower version of the software.

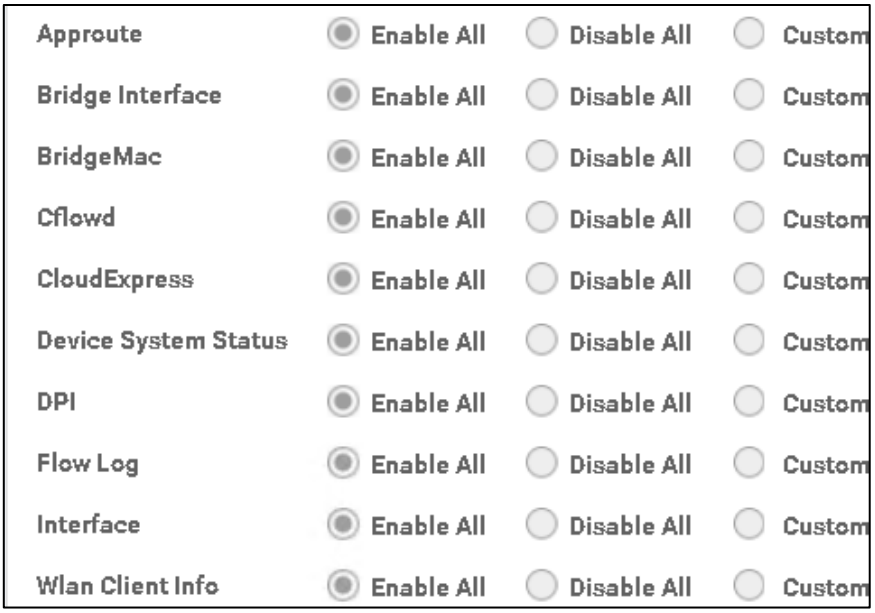
Step 115 Click **View** for **Banner**.

Enabling this setting displays a custom banner after you login to vManage.

Step 116 Click View for Statistics Setting.

 © 2020 NterOne Corporation

This allows **collections of statistics** to be enabled or disabled for devices in the network (enabled by default). Custom device selection per statistic type can be set as well.

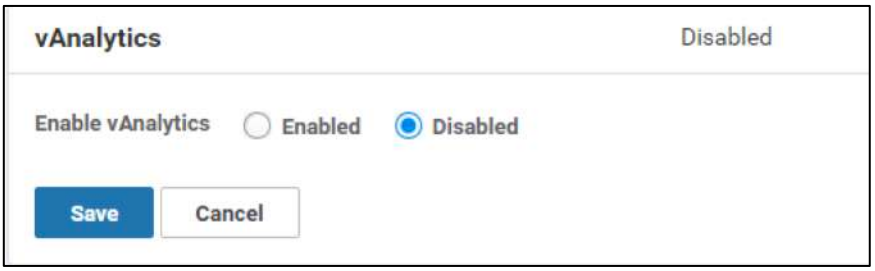


Step 117 Click **View for Cloud on Ramp for SaaS.**

To use the **Cloud onRamp for SaaS** Service this needs to be enabled (Requires subscription).

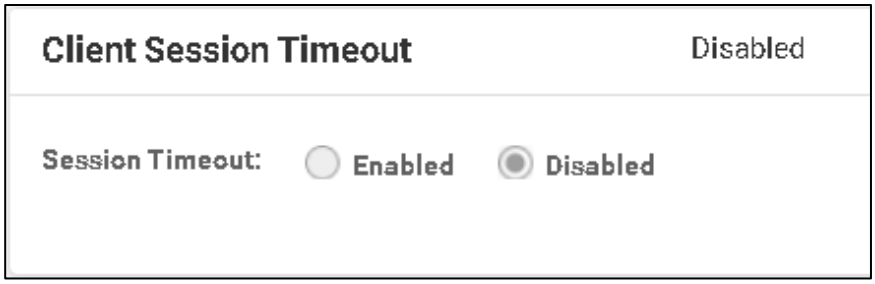
Step 118 Click **Edit for vAnalytics**.

Enable to use the vAnalytics Service (Requires subscription).



Step 119 Click **View for Client Session Timeout.**

A user's **session to vManage never times out** (default). The timeout can be adjusted in minutes.



Step 120 Click **Edit for Data Stream.**

Collecting streams of data from a network device is not enabled (default). To collect data streams, provide a host and a VPN value for collection (preferably VPN 512).

Step 121 Click **View for Tenancy Mode.**

By default, vManage operates in Single Tenancy mode (Single Organization). Multitenant mode allows multiple organizations the ability to utilize vManage. Tenancy mode needs to be set from the beginning.

When set to Multitenant mode, this setting cannot be reversed.

Once you begin to use vManage in Single Tenant mode, it's not supported to change to multitenant.

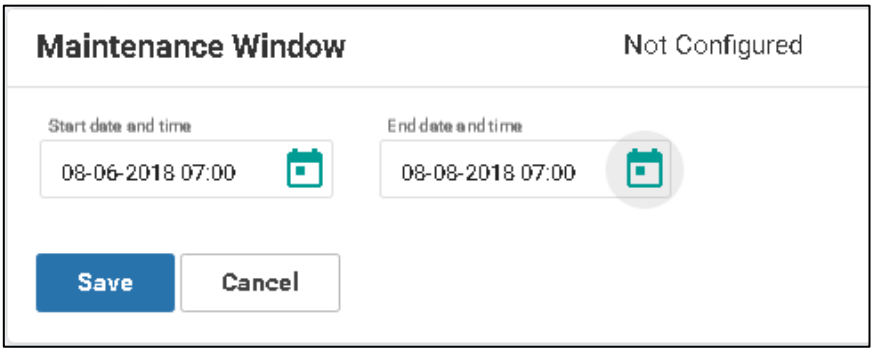
Step 122 Click **View for Statistics Configuration.**

To set the time interval at which vManage should collect statistics for devices in the network.

 © 2020 NterOne Corporation

Step 123 Click **Edit for Maintenance Window.**

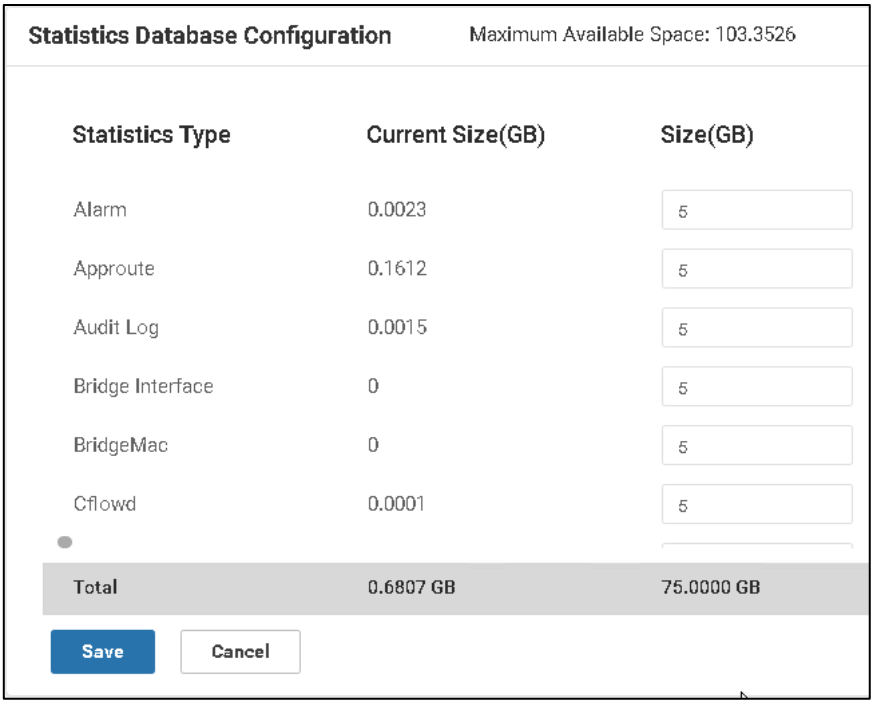
This will configure a maintenance window notification on vManage. Two days before the start of the window, the vManage Dashboard displays a maintenance window alert notification.



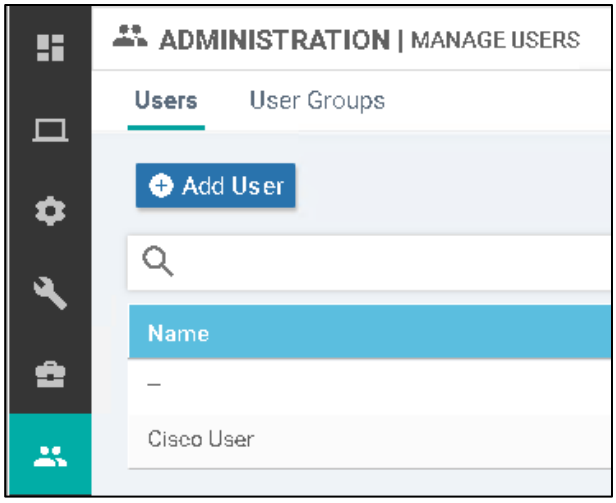
Step 124 Click **Edit for Statistics Database Configuration.**

This setting allows you to **adjust how much space is set aside for the database used to store statistics**. The size for a particular statistic type can be increased as long as it doesn't exceed the Maximum Available Space.

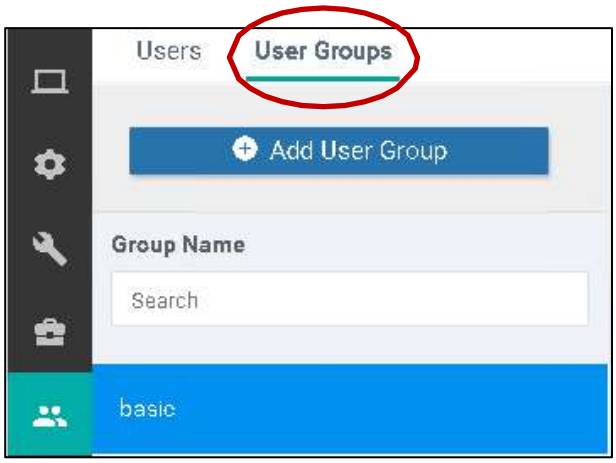
Maximum Available Space can be expanded for a vManage node.



Step 125 Go to **Administration > Manage Users.** **vManage Controller is currently in vManage mode. It will first need to be changed to CLI mode before you can add a User Group below.**



Step 126 Select **User Groups** from the menu.



Step 127 Click **+ Add User Group.**

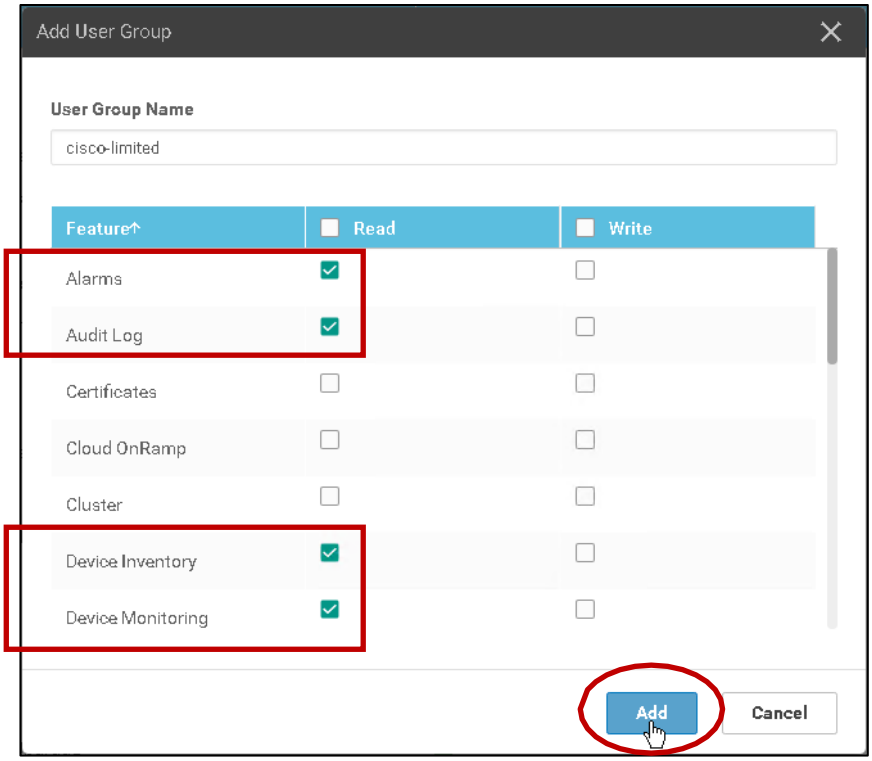
Step 128 Enter **"cisco-limited"** for the name and select the following **Read** Features.

- Alarms
- Audit Log

 © 2020 NterOne Corporation

- **Device Inventory**
- **Device Monitoring**

Step 129 Click **Add** when done.

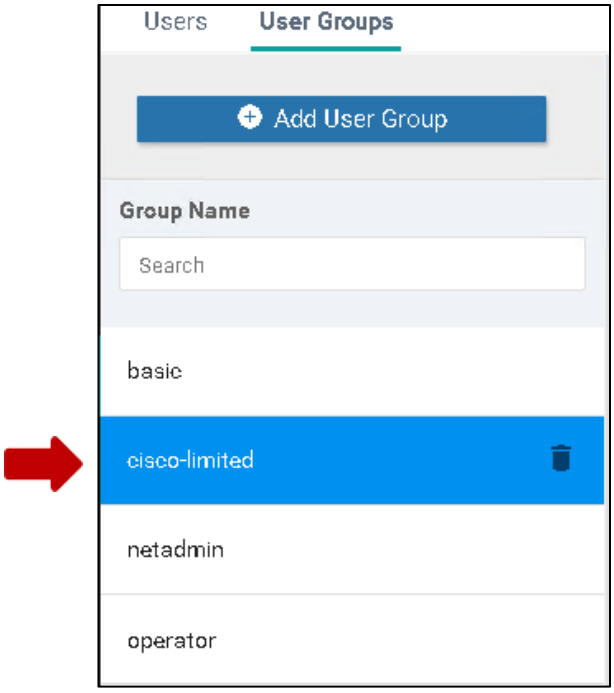


Step 130 The **Group List should be updated** with the new group. There are 3 pre-defined groups (permanent).

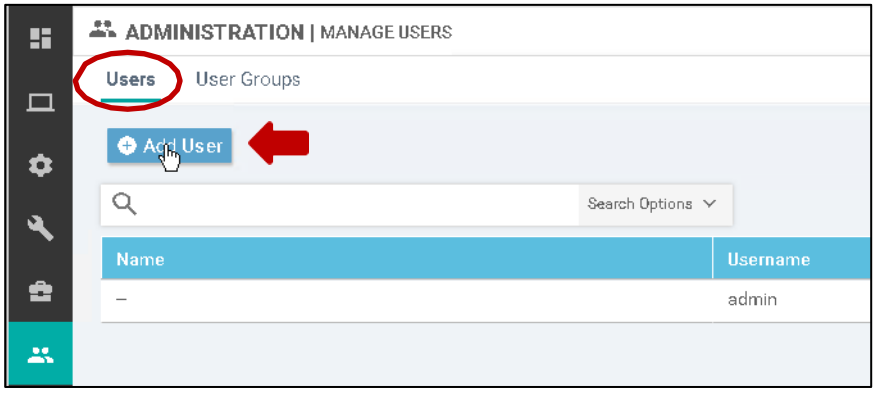
basic: Includes users who have permission to view interface and system information.

netadmin: Includes the admin user (default). The group can perform all operations on vManage.

operator: Includes users who have permission only to view information.



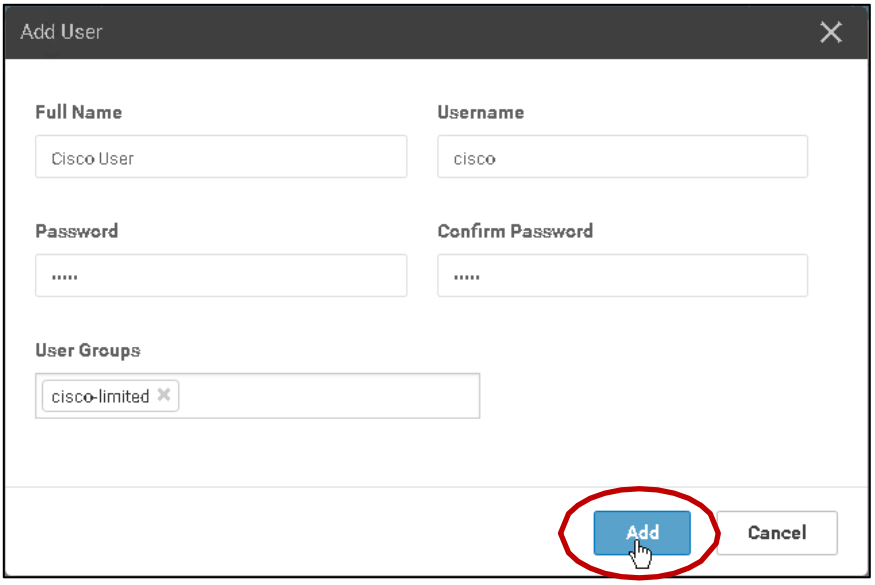
Step 131 Select **Users** from the menu and then click **+ Add User** (arrow).



Step 132 Add a **New User**. Assign the group created from the previous steps. Click **Add** when done.

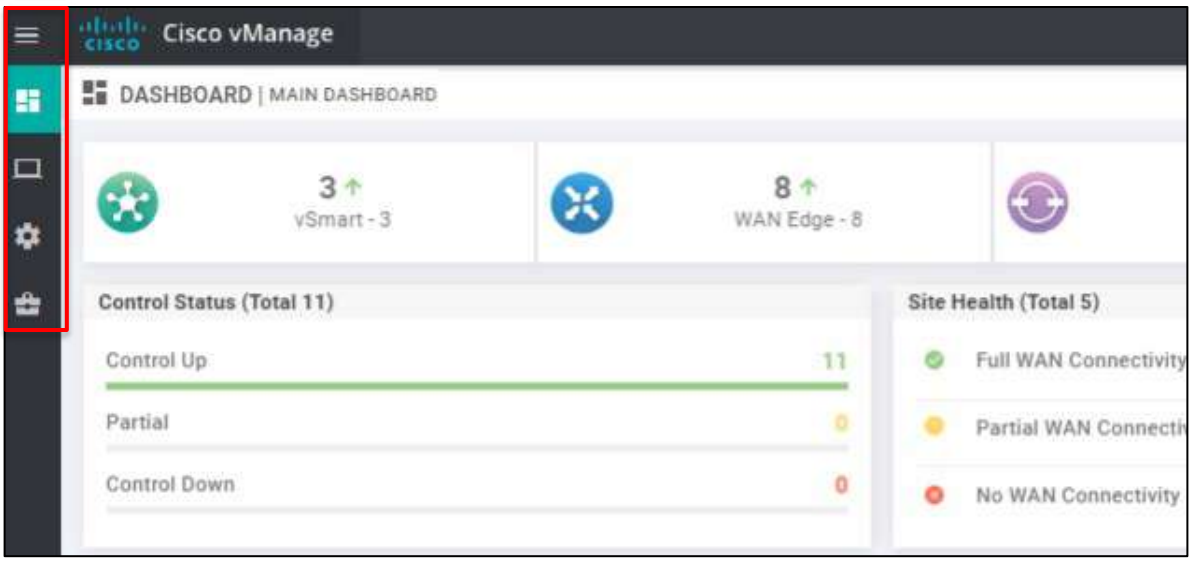
- Full Name: Cisco User
- Username: cisco
- Password: cisco
- User Group: cisco-limited

 © 2020 NterOne Corporation



Step 133 Logoff vManage **and then** login using cisco/cisco**.**

Notice that the menu options have been reduced, also in the Monitor tab there are fewer options.

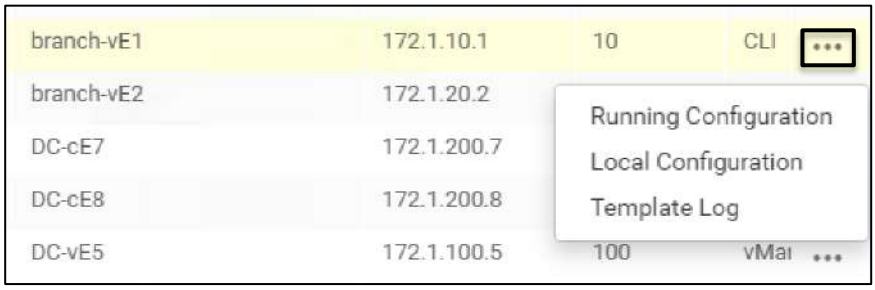


Step 134 Go to **Configuration > Devices.**

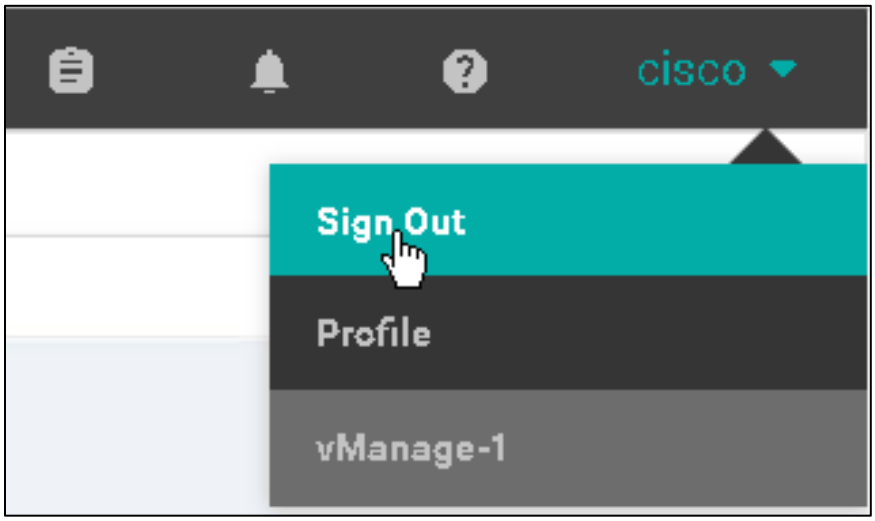
Click on the column **Hostname** to re-order the list based on alphabetical order.

Step 135 Click **…** on **branch-vE1** and notice that actions options are limited only to viewing.

These restrictions are in place as a result of the assigned User Group.



Step 136 Logoff and log back in with admin/admin. You may want to change the vManage Controller back to vManage Mode before proceeding.

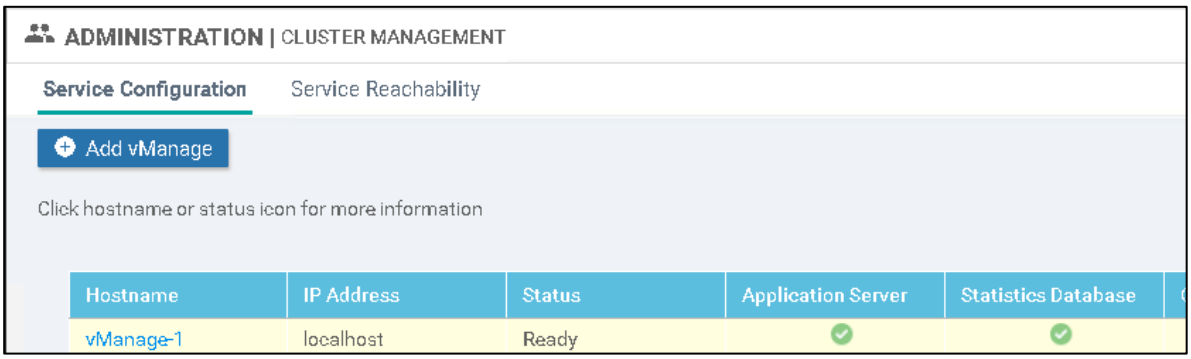


Step 137 Go to **Administration > Cluster Management.**

Use Cluster Management to create a vManage cluster. In a vManage cluster, all the cluster members communicate and work cooperatively to manage all the vBond orchestrators, vEdge routers, and vSmart controllers in the network.

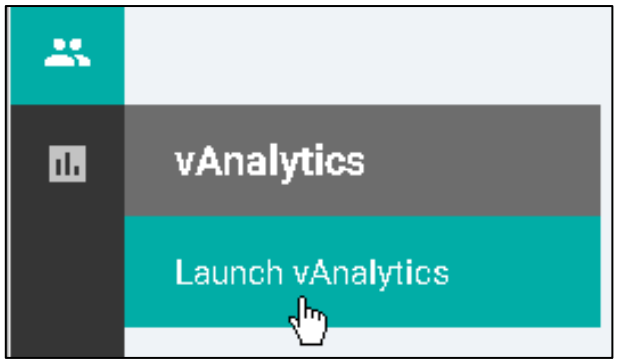
Each vManage server can manage up to about 2,700 vEdge routers in the network.

It is strongly recommended that all members of a vManage cluster be located in the same data center.



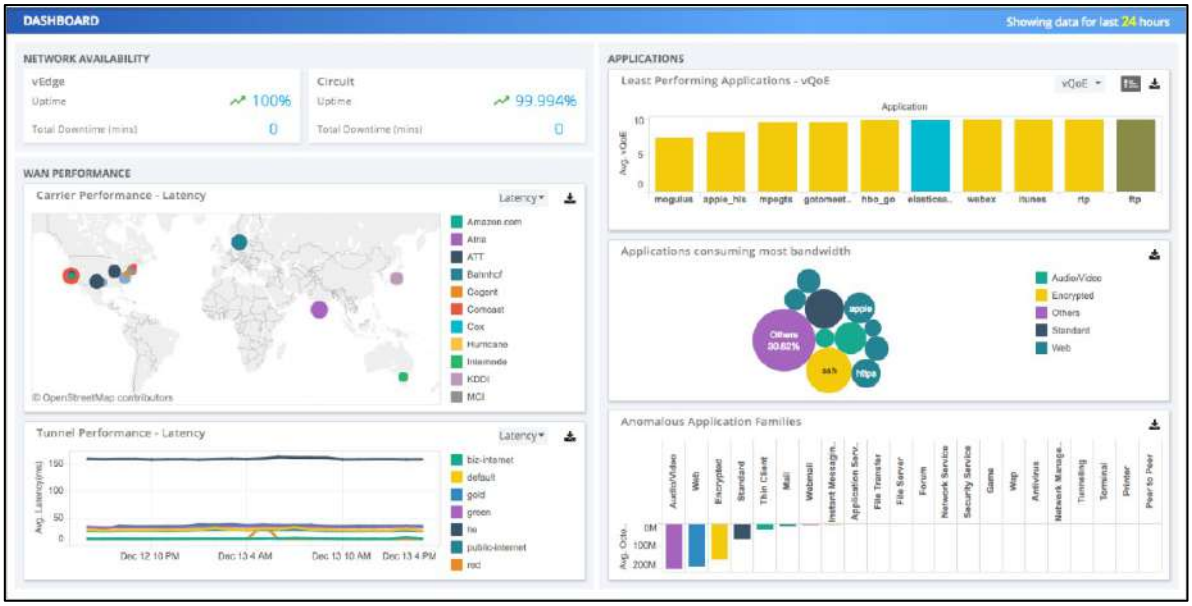
 © 2020 NterOne Corporation

Step 138 Go to **vAnalytics → Launch vAnalytics**. **(vAnalytics is not enabled). Use the following steps for information only**



Step 139 If vAnalytics has been enabled in Administration Settings and the subscription is in place, the vAnalytics Dashboard will appear.

From here you can monitor the performance of the entire network over time.



Step 140 End of Task.

Lab 7: Provision and Deploy vManage Templates

In this Lab, you will provision and deploy templates on vManage.

Activity Procedure

Perform the following steps first before proceeding to either TASK 1 or TASK 2.

Step 1 The following devices and values will be used in this Lab.

Device Name	IP Address or URL	Username	Password
vManage-1	https://1.4.28.28 – SSH 10.82.83.131	admin	admin
Branch-vE1	1.1.7.11 – SSH 10.82.83.111	admin	admin

Interface	IP Address	Default Next-Hop	Tunnel Color	VPN
ge0/1	1.1.7.11/24	0.0.0.0/0 1.1.7.22	gold	0
ge0/2	1.2.9.11/24	0.0.0.0/0 1.2.9.23	silver	0

Interface	IP Address	VPN	OSPF	Router-id
ge0/0	10.2.50.11/24	1	Area 0	172.1.10.1
loopback 1	99.99.99.99/32	1	Area 0	Passive-interface

Command	Value
host-name	branch-vE1
system-ip	172.1.10.1
site-id	10
controller-group-list	10
organization-name	NterOne Lab

Step 2 Log in to your Pod's **branch-vE1** vEdge router.

Step 3 Enter the following commands into your vEdge device CLI to remove the values from the configuration.

DO NOT REMOVE any other parameters. The vEdge router will be re-configured using Templates.

 © 2020 NterOne Corporation

```
branch-vE1# config
 Entering configuration mode terminal
 branch-vE1(config)# no vpn 1
 branch-vE1(config)# vpn 0
 branch-vE1(config-vpn-0)# no interface ge0/2
 branch-vE1(config-vpn-0)# no ip route 0.0.0.0/0 1.2.9.23
 branch-vE1(config-vpn-0)# exit
 branch-vE1(config)# omp
 branch-vE1(config-omp)# no advertise ospf
 branch-vE1(config-omp)# no advertise connected
 branch-vE1(config-omp)# no advertise static
 branch-vE1(config-omp)# shutdown
 branch-vE1(config-omp)# exit
 branch-vE1(config)# commit
 Commit complete.
 branch-vE1(config)# end
```

Step 4 Use the command **show run** to verify the configuration. The system parameters will remain the same.

```
branch-vE1# show run
 system
  host-name               branch-vE1
  system-ip               172.1.10.1
  site-id                 10
  admin-tech-on-failure
  no route-consistency-check
  organization-name       "NterOne Lab"
  vbond vbond.demo.local
  aaa
   auth-order local radius tacacs
   usergroup basic
    task system read write
    task interface read write
   !
   usergroup netadmin
   !
   usergroup operator
    task system read
    task interface read
    task policy read
```

```
    task routing read
    task security read
   !
   usergroup tenantadmin
   !
   user admin
    password
$6$3cM/dwNf027gvKv5$o6KqKA5SagTUvKltEpnaq8hg/EhEf8dh3crUvfxRNtPkJmRoNiaR
UpqMBpmlqoXUSmPvyxJYMhFxjkkcwAhEn.
   !
  !
  logging
   disk
    enable
   !
  !
 !
 omp
  shutdown
  graceful-restart
 !
 security
  ipsec
   authentication-type ah-sha1-hmac sha1-hmac
  !
 !
 vpn 0
  dns 1.4.28.89 primary
  interface ge0/1
   ip address 1.1.7.11/24
   tunnel-interface
    encapsulation ipsec
    color gold
    allow-service all
    no allow-service bgp
    allow-service dhcp
    allow-service dns
    allow-service icmp
    no allow-service sshd
    no allow-service netconf
    no allow-service ntp
```

 © 2020 NterOne Corporation

```
    no allow-service ospf
    no allow-service stun
    allow-service https
   !
   no shutdown
  !
  ip route 0.0.0.0/0 1.1.7.22
 !
 vpn 512
  interface eth0
   ip address 10.82.83.111/20
   no shutdown
  !
 !
```

Step 5 Use the following commands to verify the configuration. There should be no **OMP peers** and no **OMP routes** (**vRoutes),** as the Overlay network has been removed and OMP is disabled. **Only one tunnel remains "gold" which provides Control-Plane connectivity** so that vManage and the controllers can still maintain visibility into the device.

DO NOT REMOVE anything else.

```
branch-vE1# show omp peers

branch-vE1# show ip route                    (Formatted for clarity)
Codes Proto-sub-type:
  IA -> ospf-intra-
  area, IE -> ospf-
  inter-area, E1 ->
  ospf-external1, E2 ->
  ospf-external2,
  N1 -> ospf-nssa-external1, N2
  -> ospf-nssa-external2, e ->
  bgp-external, i -> bgp-
  internal
Codes Status flags:
  F -> fib, S ->
  selected, I ->
  inactive, B ->
  blackhole, R ->
  recursive
```

```
                                                PROTOCOL  NEXTHOP  NEXTHOP
VPN              PREFIX            PROTOCOL  SUB TYPE  IF NAME  ADDR    VPN  TLOC
IP COLOR EN STATUS
-------------------------------------------------------------------------------
-------
 0      0.0.0.0/0          static       -    ge0/1    1.1.7.22  -  -    -     -    F,S
 0      1.1.7.0/24         connected    -    ge0/1    -             -   -     -    F,S
 0      172.1.10.1/32      connected   -     system   -             -   -     -    F,S

          --- VPN 512 OMITTED ---
```

branch-vE1# **show control connections**
(*Formatted for clarity) (Output may vary)*

```
                                                   PEER                   PEER
PEER    PEER PEER         SITE  DOMAIN  PEER       PRIV PEER              PUB  LOCAL  STATE
TYPE    PROT SYSTEM IP    ID    ID      PRIVATE IP PORT PUBLIC IP         PORT COLOR
-----------------------------------------------------------------------------------------
vsmart  dtls 172.1.0.16  1011  1      1.3.26.26    1234 1.3.26.26       12346 gold   up
                                                     6
vsmart  dtls 172.1.0.19  1013  1      1.3.29.29    1234 1.3.29.29       12346 gold   up
                                                     6
vmanage dtls 172.1.0.18  1001  0      1.4.28.28    1234 1.4.28.28       12346 gold   up
                                                     6
```

branch-vE1# **show control local-properties**

```
personality                          vedge
sp-organization-name                 "NterOne Lab"
organization-name                    "NterOne Lab"
certificate-status                   Installed
root-ca-chain-status                 Installed

certificate-validity                 Valid
certificate-not-valid-before Nov 26 04:20:15 2017 GMT
certificate-not-valid-after          Nov 24 04:20:15 2027 GMT

dns-name                             vbond.demo.local
site-id                              10
domain-id                            1
protocol                             dtls
tls-port                             0
system-ip                            172.1.10.1
chassis-num/unique-id                34bf864b-
8cf6-0b57-2489-c74b03d212f4 serial-num  6F7B918C
token                                Invalid
vsmart-list-version                  4
keygen-interval                      1:00:00:00
```

 © 2020 NterOne Corporation

```
retry-interval                          0:00:00:18
no-activity-exp-interval                0:00:00:12
dns-cache-ttl                           0:00:02:00
port-hopped                             TRUE
time-since-last-port-hop                0:00:08:47
number-vbond-peers                      0
number-active-wan-interfaces 1



NAT TYPE: E -- indicates End-point independent
          mapping A -- indicates Address-port
          dependent mapping N -- indicates Not
          learned
          Note: Requires minimum two vbonds to learn the NAT type


                                                                                                  RESTRICT/
          PUB                PUB     PRIVATE   PRV    PRV                          MAX   CONTROL/
          INTERFACE IPV4     PORT    IPV4      IPV6 PORT   VS/VM COLOR STATE CNTRL  STUN      LR/LB
          ------------------------------------------------------------------------------------
          ----
          ge0/1              1.1.7.11 12366  1.1.7.11  :: 12366    2/1  gold  up     2
                             no/yes/no       No/No
```

Note	**Choose TASK 1 to create templates using Feature Templates (recommended).**

Note	**Choose TASK 2 to create templates using CLI Templates.**

Task 1: Deploy Feature-Based Device Template

In this Task you will deploy a Device Template using Feature Templates.

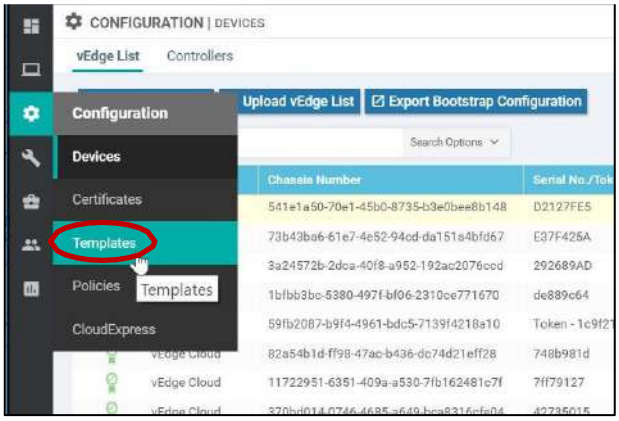
Activity Procedure

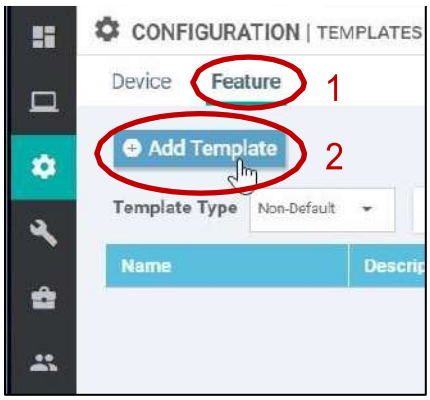
Complete these steps:

Step 1 **Log in to vManage.** You will use vManage to create templates.

Step 2 **Feature Templates** will be created first, followed by a **Device Template** that will use them.

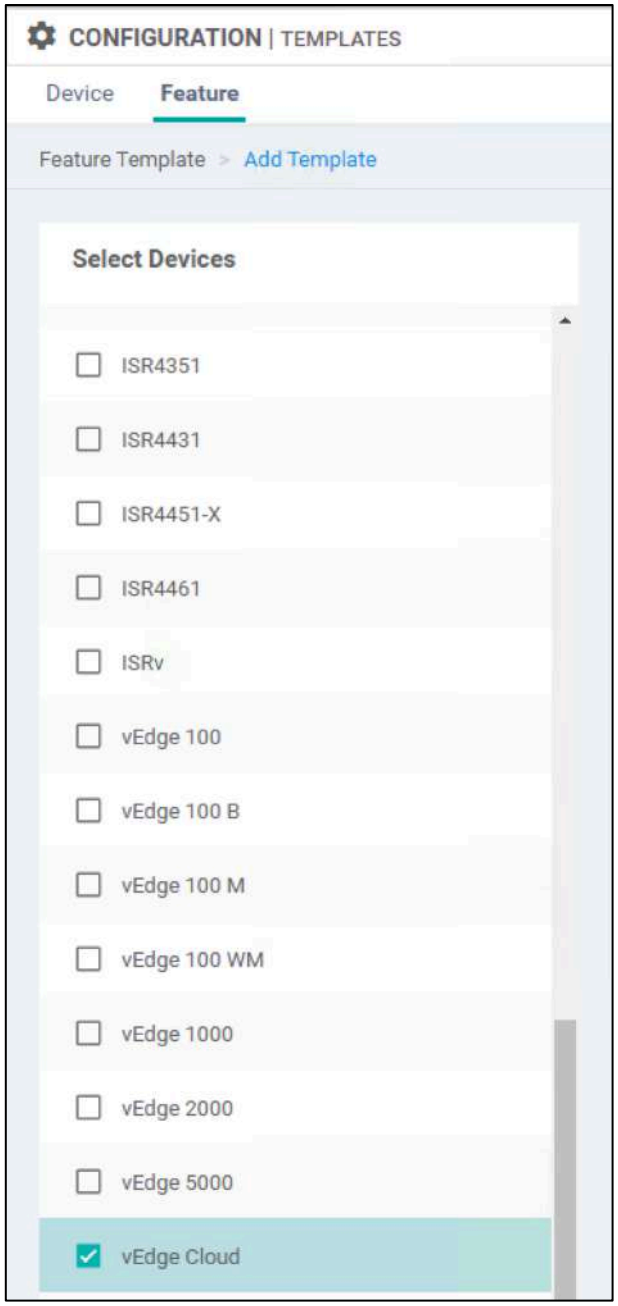
Step 3 Go to **Configuration > Templates** and select the **Feature** tab**,** and then click **+ Add Template** from the top left corner of the templates window.



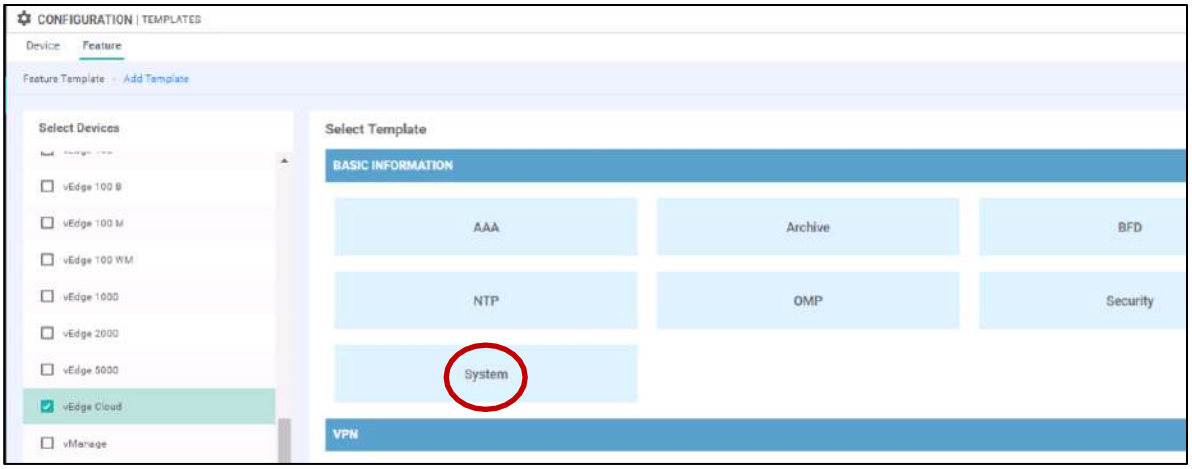


Step 4 **Scroll down to select vEdge Cloud from the Select Devices menu**.

 © 2020 NterOne Corporation



Step 5 Select **System** from the Select Template window.



Step 6 The following names can be used for the feature templates. This is a list of the feature templates you will be configuring in Steps 7 and beyond.

Template	Template Name	Value(s)
System	vEdge-Cloud-System-Template	172.1.10.1 - Site 10 - Group 10
OMP	vEdge-Cloud-OMP-Template	
VPN (0)	vEdge-Cloud-VPN0-Template	0.0.0.0/0 (1.1.7.22 - 1.2.9.23) dns 1.4.28.89
VPN (1)	vEdge-Cloud-VPN1-Template	
VPN (512)	vEdge-Cloud-VPN512-Template	
VPN-Interface-Ethernet (0)	vEdge-Cloud-Gold-Interface-Template	ge0/1 – tunnel – gold – ipsec
VPN-Interface-Ethernet (0)	vEdge-Cloud-Silver-Interface-Template	ge0/2 – tunnel – silver – ipsec
VPN-Interface-Ethernet (1)	vEdge-Cloud-LAN-Interface-Template	ge0/0
VPN-Interface-Ethernet (1)	vEdge-Cloud-Loopback1-Interface-Template	loopback1 - (OSPF Passive-Interface)
VPN-Interface-Ethernet (512)	vEdge-Cloud-Management-Interface-Template	eth0 – 10.82.83.111/20
OSPF	vEdge-Cloud-OSPF-Template	area 0 – ge0/0 and loopback1
SNMP	vEdge-Cloud-Snmp-Template	
Banner	vEdge-Cloud-Banner-Template	

 © 2020 NterOne Corporation

Step 7 Change the **Scope of a parameter value** using one of the options below.

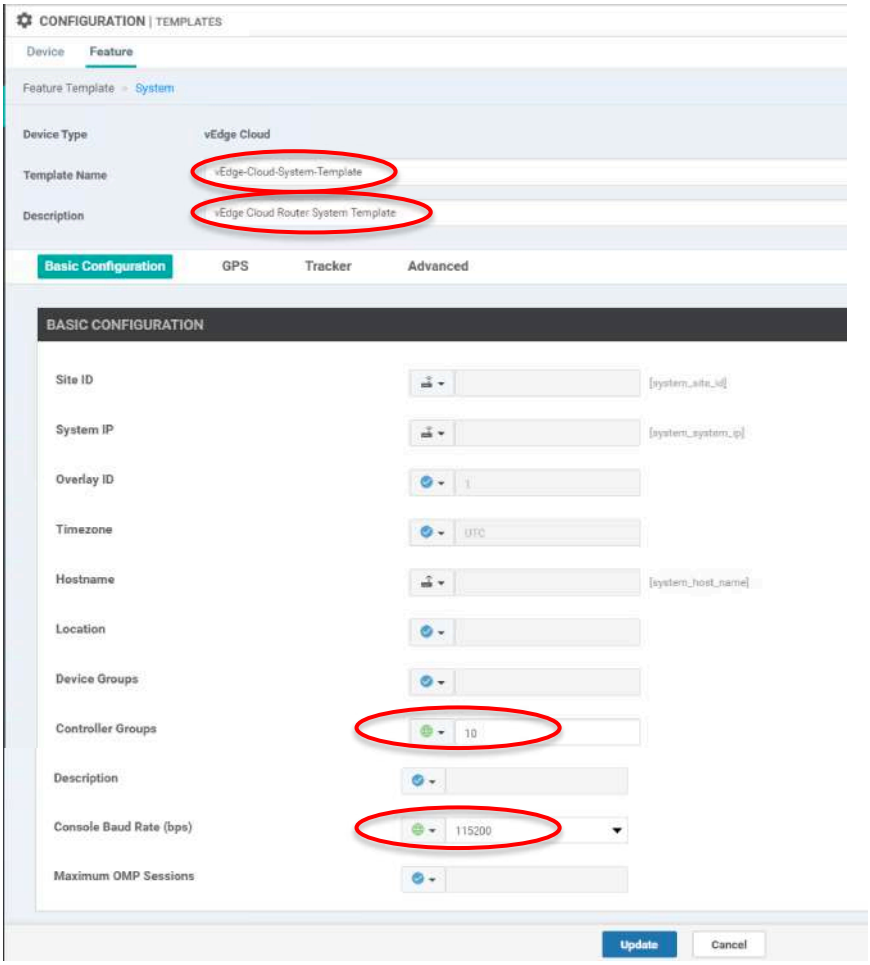
Global - If the parameter value is global then **every** device using this template will receive the value.

Device Specific - A value needs to be specified when the template is attached to a device.

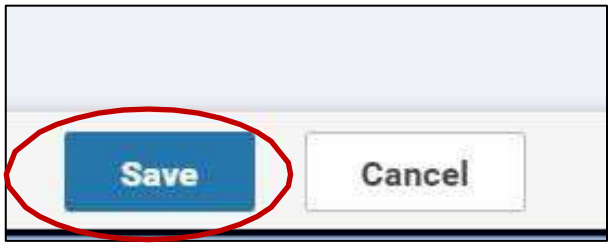
Default Value - The system will use a default value for the parameter.

Step 8 Enter the Template Name **"vEdge-Cloud-System-Template"** and "**Description"** (Descriptions are required). Template **"names"** can be found in the Table above in Step 6.

Use **Global** and enter the value **"10"** for the parameter **Controller Groups**. Use **Global** and enter **115200** for the parameter **Console Baud Rate (bps)**. Leave all the other values as they are.



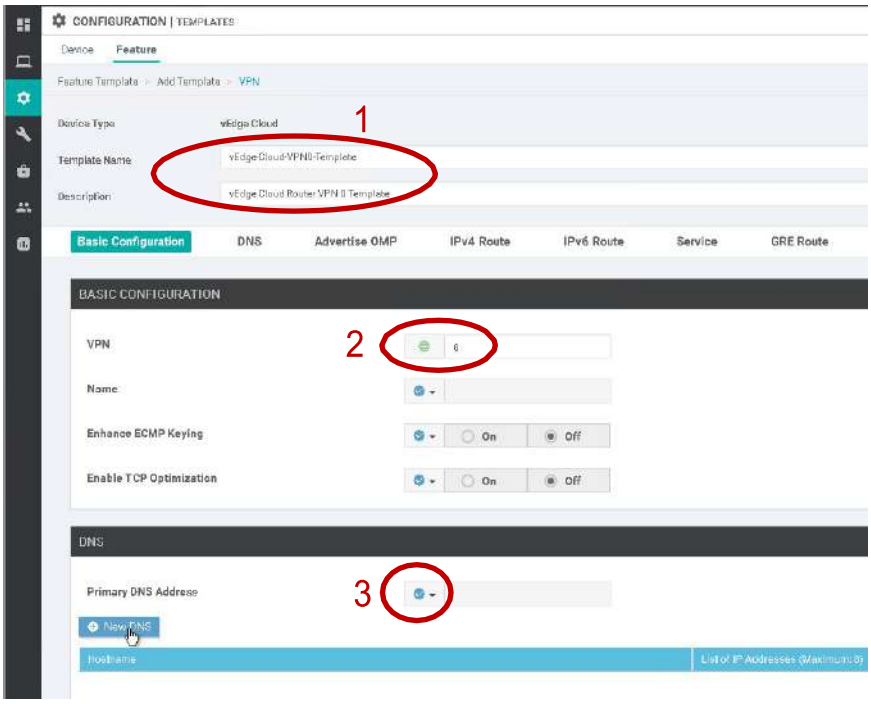
Step 9 Click **Save** at the bottom of the feature window.



Step 10 If not already there, go to **Configuration > Templates > Feature** and select **+Add Template** Select **vEdge Cloud** from the Select Devices menu and then on the Template matrix click **VPN**. For Template Name enter **"vEdge-Cloud-VPN0-Template"** and enter a D**escription**.

Select **Global** for the **VPN** parameter and enter **"0".**

Select Global for Primary DNS Address.



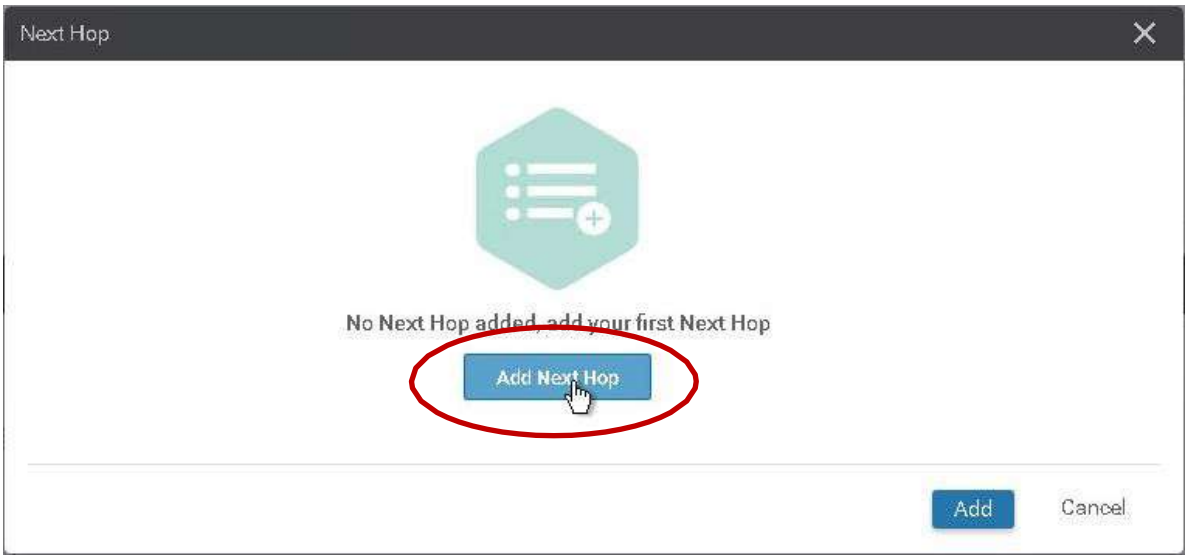
Step 11 **Enter** "1.4.28.89" **for** Primary DNS Address (primary DNS server)**.**

Step 12 Select the **IPv4 Route** tab from the Feature menu and then click the **+ New IPv4 Route.**

 © 2020 NterOne Corporation

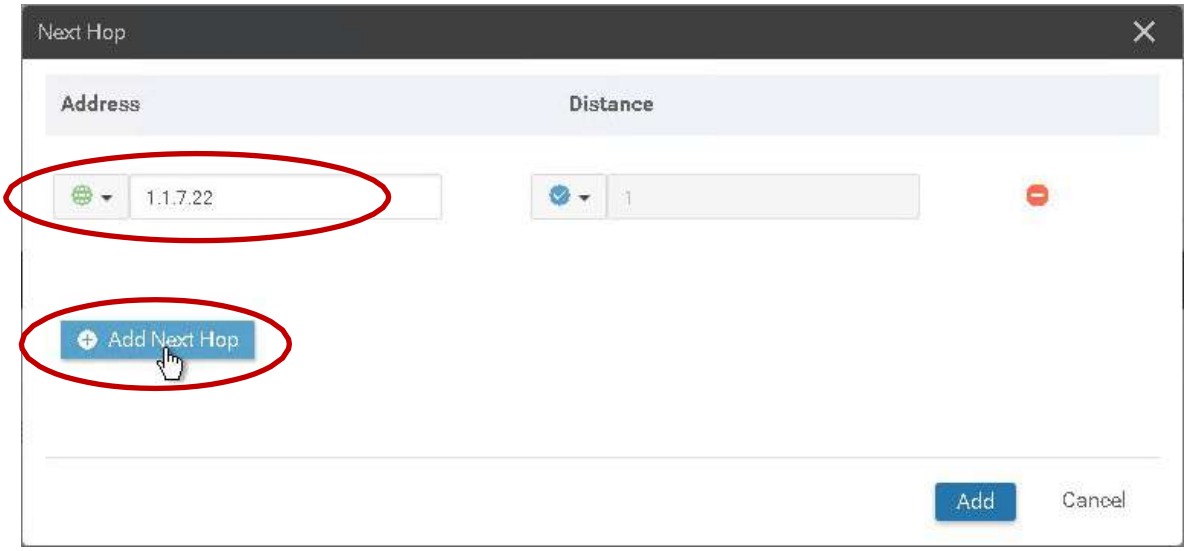
Step 13 In the Prefix field select Global and enter “0.0.0.0/0” then click + Add Next Hop.

Step 14 Click Add Next Hop.



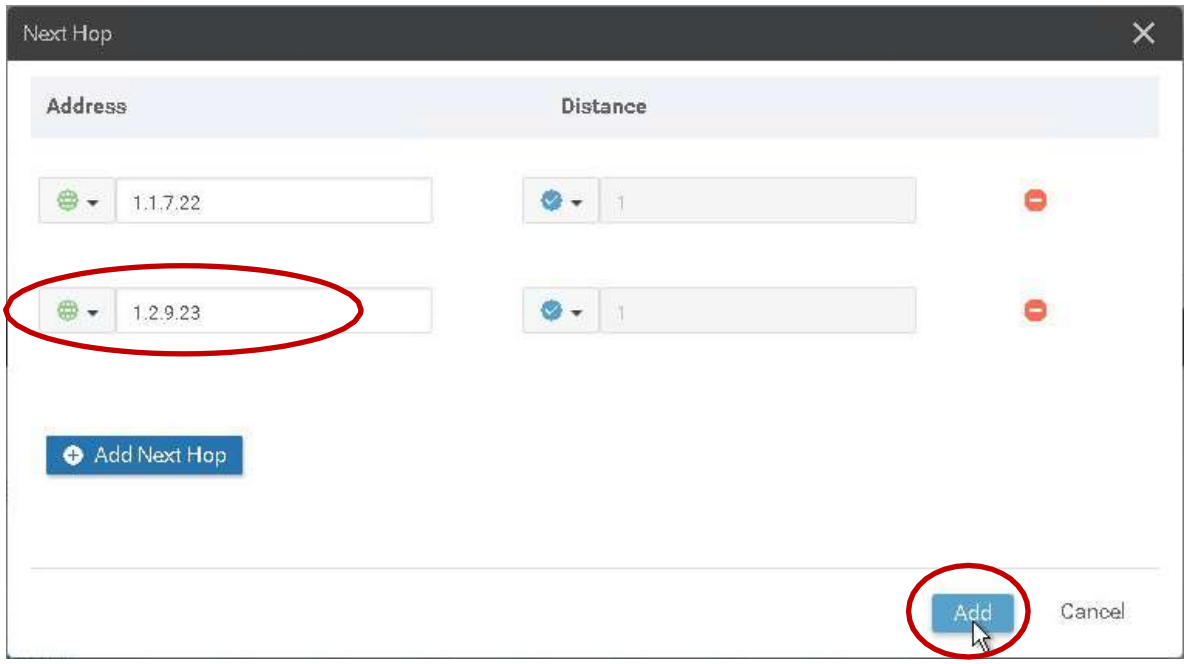
Step 15 In the Address field select Global and enter “1.1.7.22” (Router for the Gold transport)

Click **+ Add Next Hop** to add the second gateway.



Step 16 In the Address field select Global and enter "1.2.9.23". (Router for the Silver transport)

Click **Add** when done.



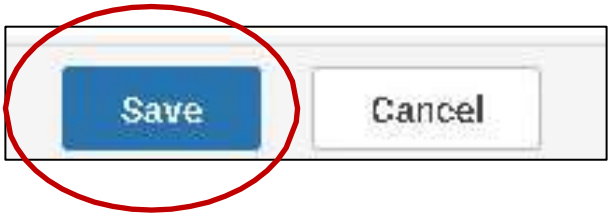
Step 17 Verify Prefix reads **0.0.0.0/0** and Next Hop indicates **2 Next Hop.** Go back and correct if necessary.

Click **Add**.

 © 2020 NterOne Corporation

Step 18 Once the 2 Gateways have been added, your IPv4 Route category should look similar to the following.

Step 19 Click **Save** to save the VPN template.

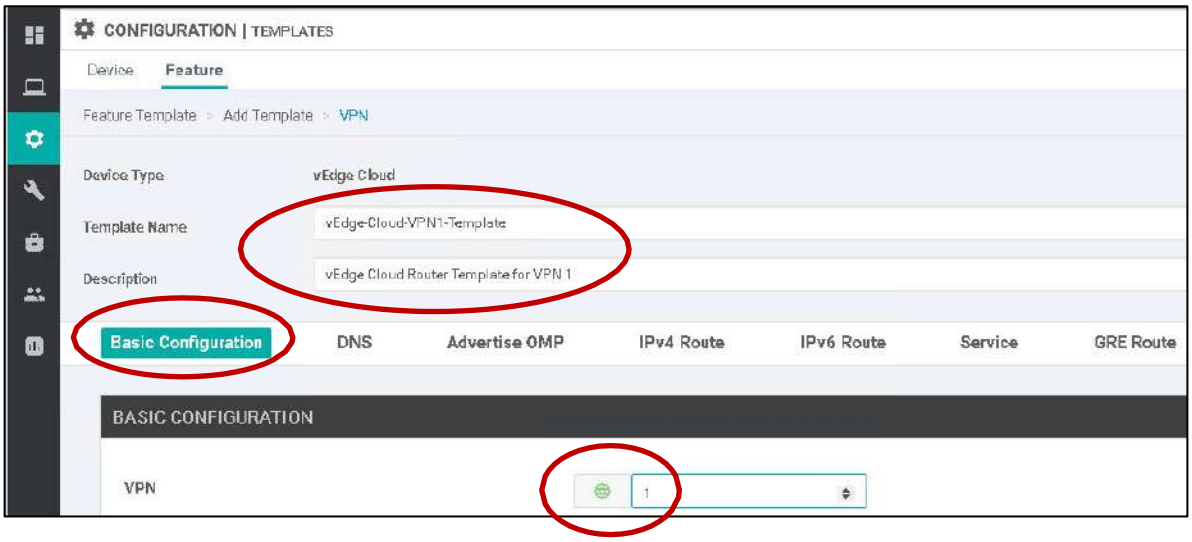


Step 20 Go to **Configuration > Templates > Feature and select +Add Template**

Select **vEdge Cloud** and then on the Template matrix click **VPN**.

For Template Name enter **"vEdge-Cloud-VPN1-Template"** and enter a **Description**. Select **Global** for the **VPN** parameter and enter **"1"**

Click **Save** when done.

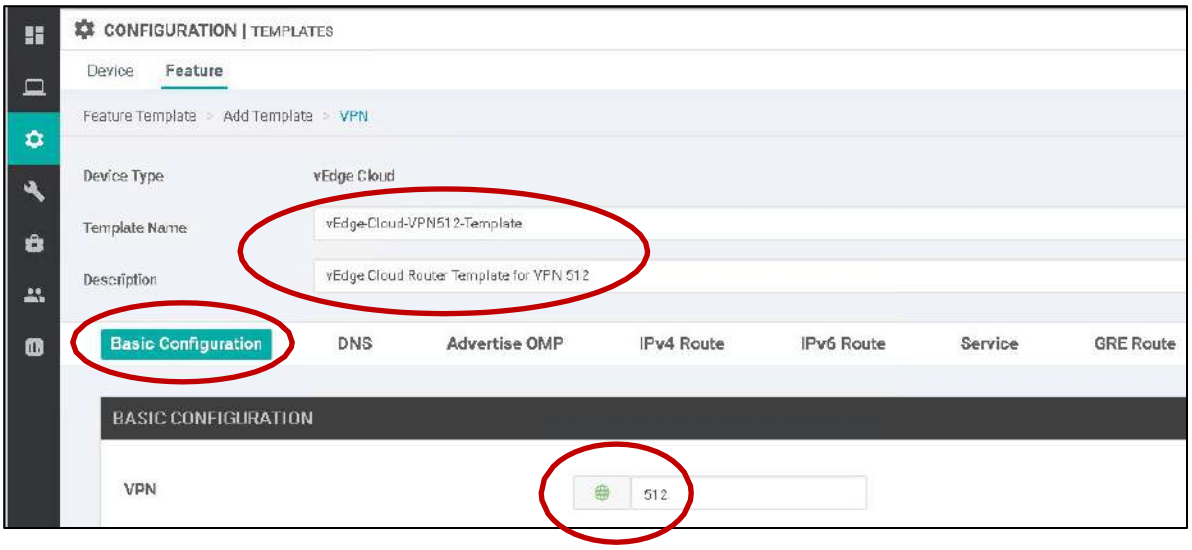


Step 21 Go to **Configuration > Templates > Feature and select +Add Template**

Select vEdge Cloud and then on the Template matrix click VPN.

For Template Name enter "**vEdge-Cloud-VPN512-Template**" and enter a Description. Select Global for the VPN parameter and enter "512"

Click Save when done.



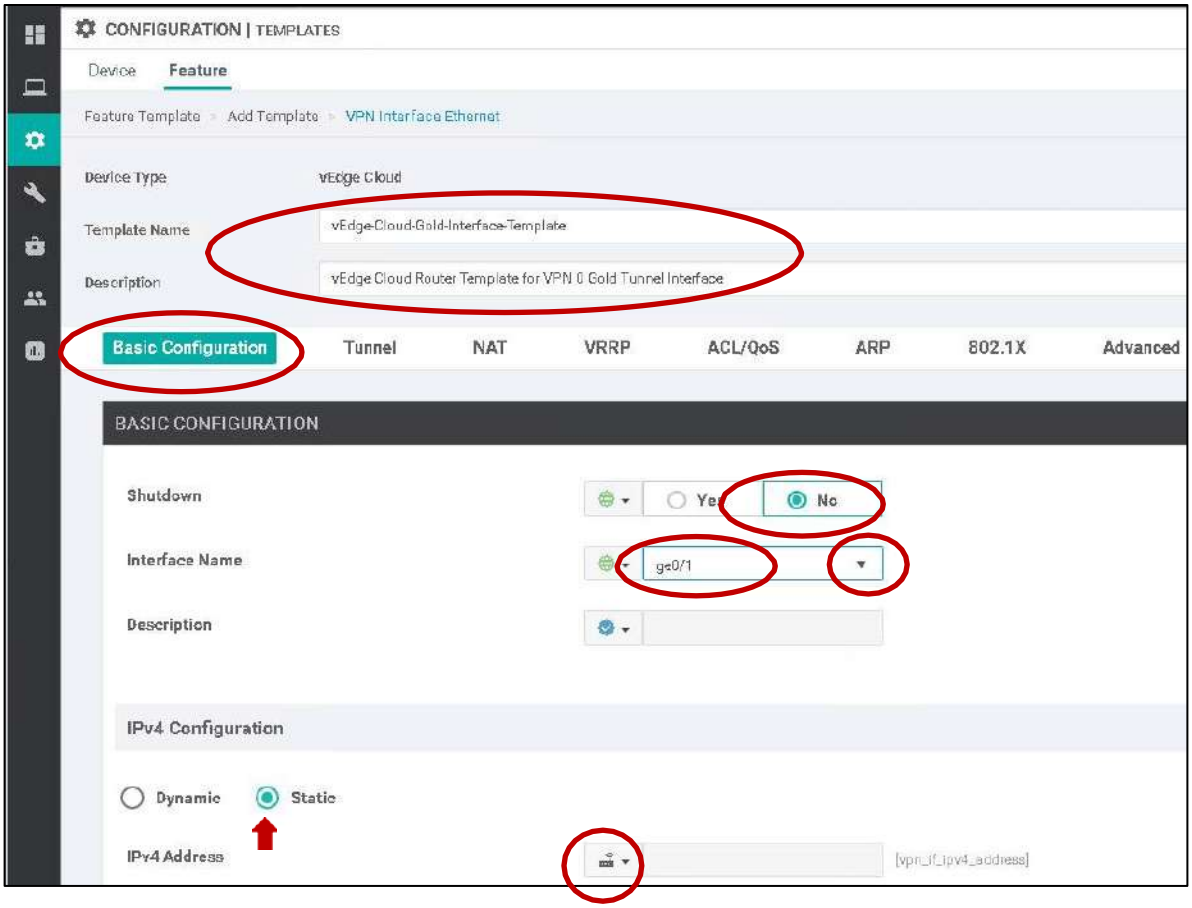
Step 22 Go to **Configuration > Templates > Feature and select +Add Template**

Select **vEdge Cloud** and then on the Template matrix click **VPN-Interface-Ethernet**.

For Template Name enter **"vEdge-Cloud-Gold-Interface-Template"** and enter a **Description**. Select **Global** for the **Shutdown** parameter and then select **No**.

Select **Global** for the **Interface Name** and in the drop-down select **"ge0/"**. Enter **"1"** manually after for ge0/1. Select **Static** for IPv4 Configuration and then select **Device Specific** for the IPv4 Address

 © 2020 NterOne Corporation



Step 23 Click on **Tunnel** from the categories menu.

Use **Global** for the **Tunnel Interface** parameter, and then select **On.**

Use **Global** for the **Color** parameter and then select **gold** from the list. Click **Save** when done.

Step 24 **Repeat steps 22 and 23 to create another VPN-Interface-Ethernet but this time use "silver" for the color and "ge0/2" for the interface. Make sure you don't forget to "no shut" the interface and set device specific for the ipV4 address.**

Step 25 In previous steps, the WAN transport Tunnel interfaces **gold** and **silver** were created. These interfaces will also carry control-plane traffic. In the next

steps, two additional interfaces will be created. These interfaces represent the Internal Network interfaces (Site local / Service Side).

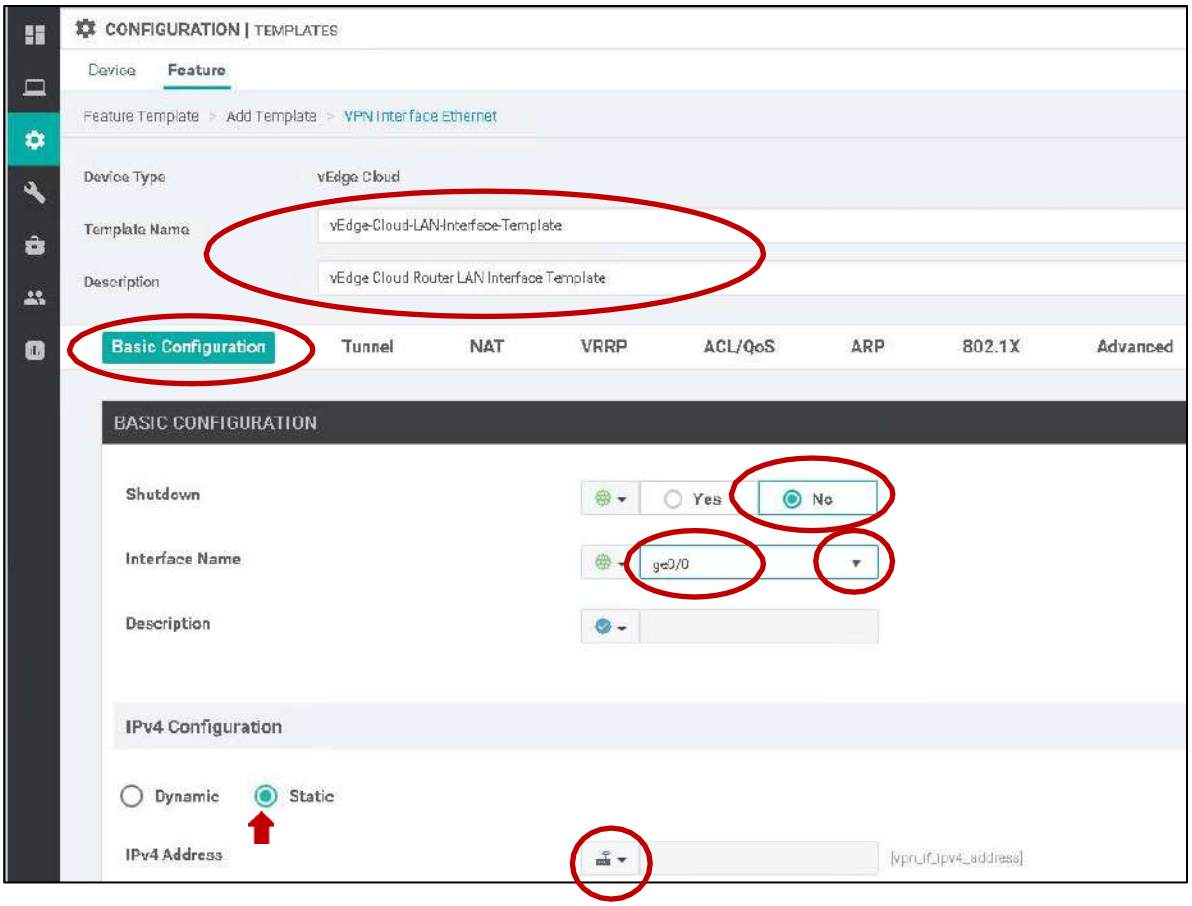
Step 26 Go to Configuration > Templates > Feature and select +Add Template

Select **vEdge Cloud** and then on the Template matrix click **VPN-Interface-Ethernet**.

For Template Name enter **"vEdge-Cloud-LAN-Interface-Template"** and enter a **Description**. Select **Global** for the **Shutdown** parameter and select **No**.

Select **Global** for the **Interface Name** and in the drop-down select **"ge0/"**. Enter **"0"** manually after for ge0/0. Select **Static** for IPv4 Configuration and then select **Device Specific** for the IPv4 Address

Click **Save** when done.



Step 27 Go to Configuration > Templates > Feature and select +Add Template

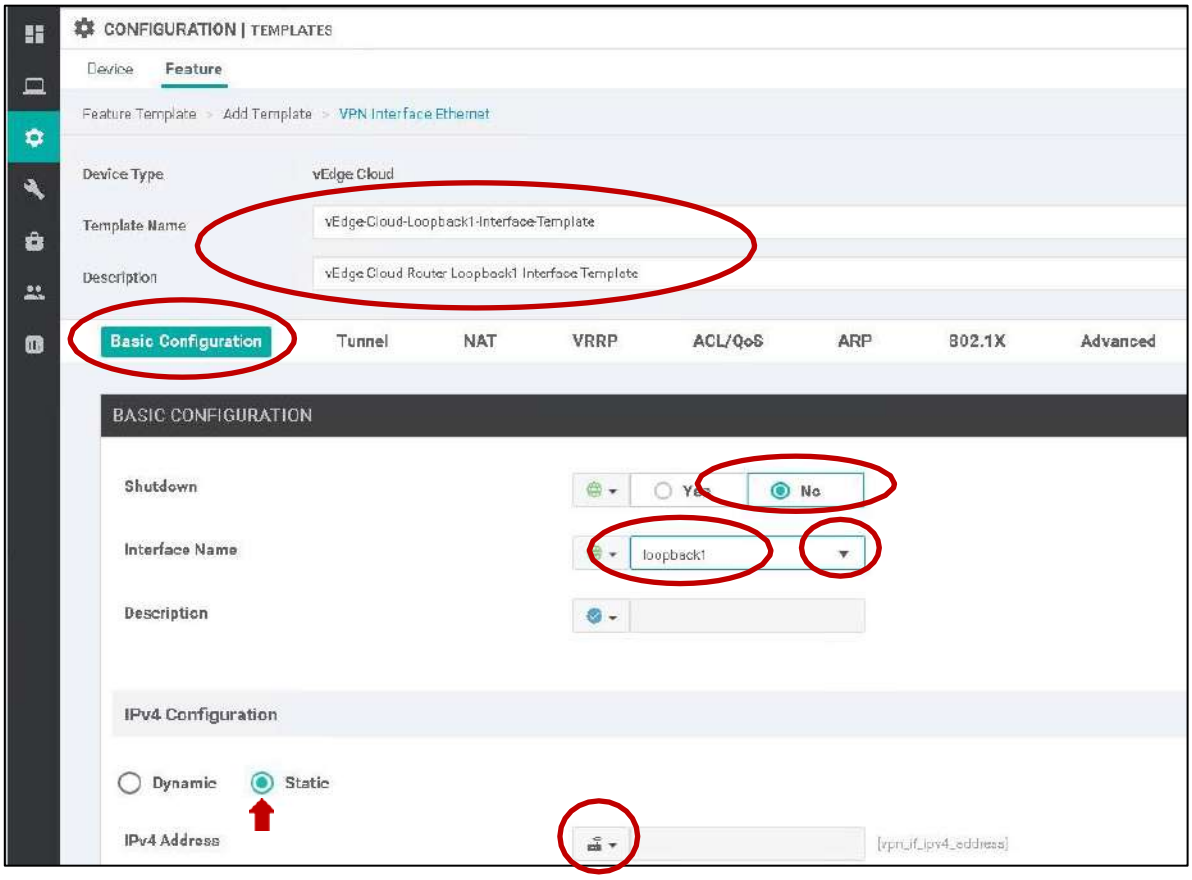
Select **vEdge Cloud** and then on the Template matrix click **VPN-Interface-Ethernet**.

For Template Name enter **"vEdge-Cloud-Loopback1-Interface-Template"** and enter a **Description**. Use **Global** for the **Shutdown** parameter and select **No**.

 © 2020 NterOne Corporation

Use **Global** for the **Interface Name** and in the drop-down select **"loopback"**. Enter **"1"** manually after for loopback1. Select **Static** for IPv4 Configuration and then select **Device Specific** for the IPv4 Address

Click **Save** when done.



Step 28 Go to Configuration > Templates > Feature and select +Add Template Select vEdge Cloud and then on the Template matrix click VPN-Interface-Ethernet.

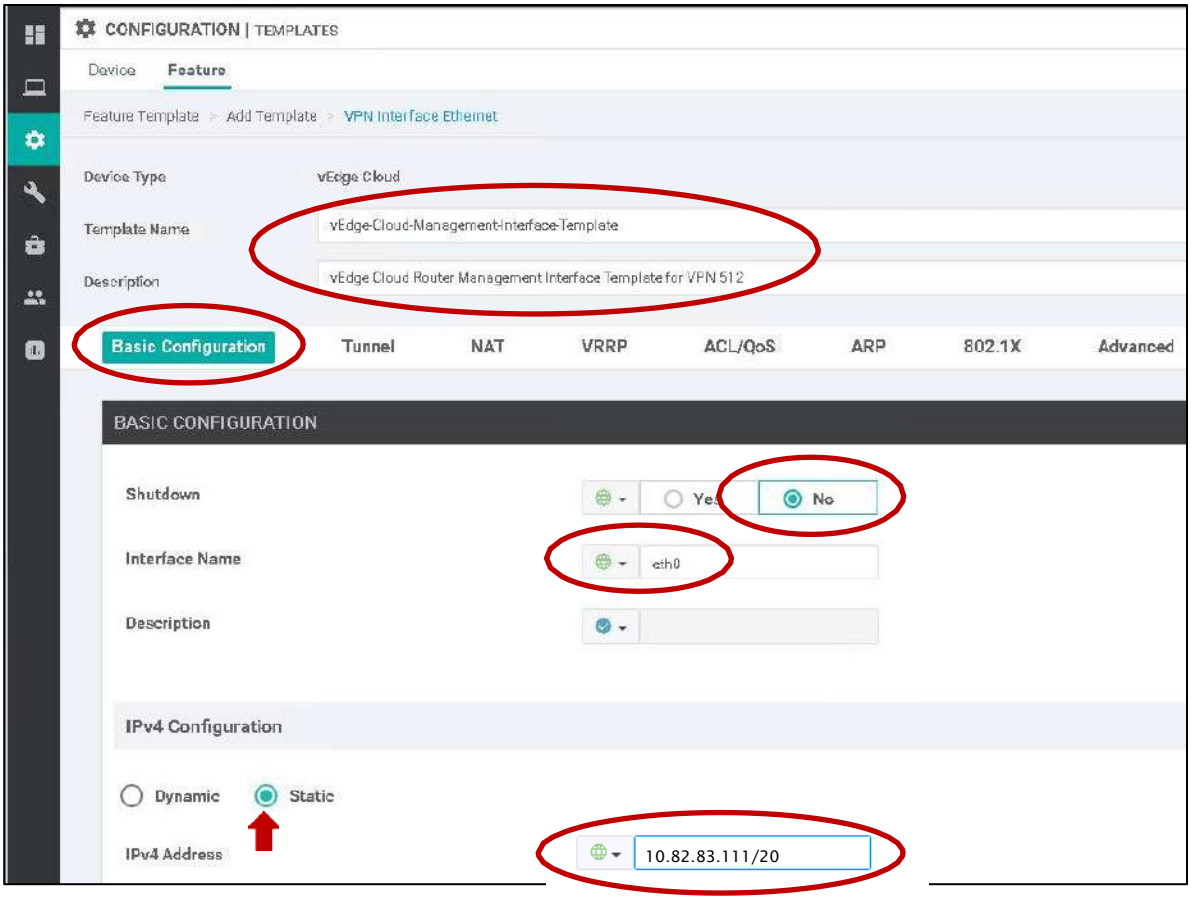
For Template Name enter **"vEdge-Cloud-Management-Interface-Template"** and enter a **Description**. Select **Global** for the **Shutdown** parameter and select **No**.

Select **Global** for the **Interface Name** and in the drop-down select **"eth"**. Enter **"0"** manually after for eth0.

For IPv4 select **Static**, then use **Global**, enter **"10.82.83.111/20"**

STOP and verify that the IPV4 Configuration is Correct. (You will be disconnected when you attach the vEdge to the Device Template, if it is not).

Click **Save** when done.

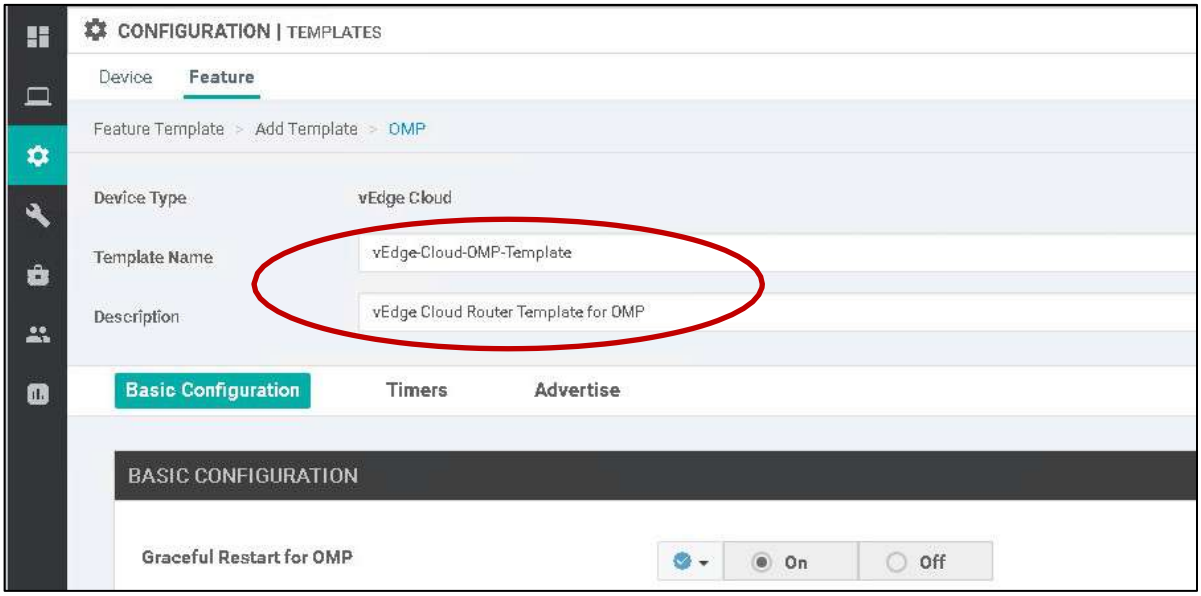


Step 29 Go to Configuration > Templates > Feature and select +Add Template

Select **vEdge Cloud** and then on the Template matrix click **OMP**.

For Template Name enter **"vEdge-Cloud-OMP-Template"** and enter a **Description**. Go to the **Next Step.**

 © 2020 NterOne Corporation



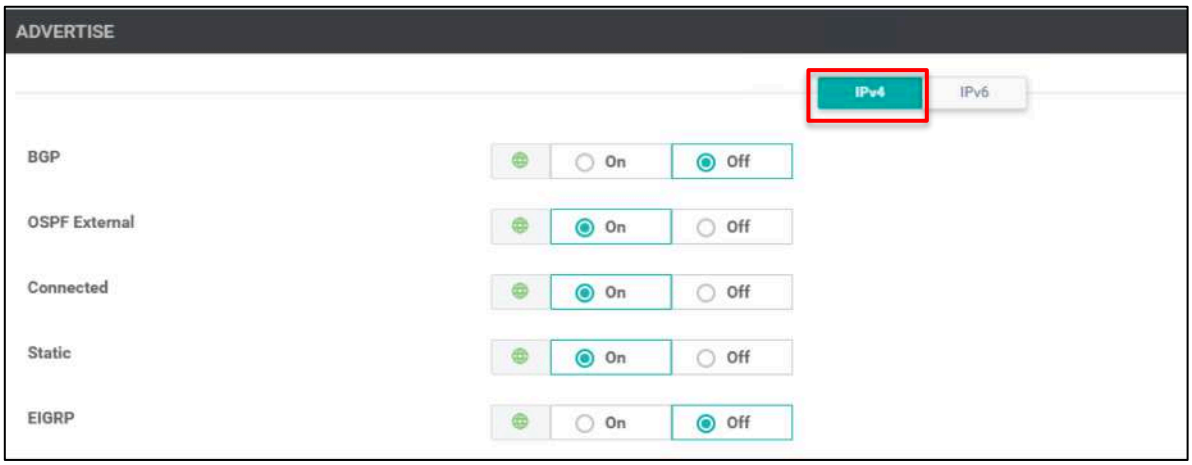
Step 30 Click **Advertise**. This will redirect you to the "Advertise" section of the OMP window.

Click the IPv4 button and select **On** for the following:

OSPF External

Connected

Static



Click the IPv6 button and select **Off** for all the parameters listed:

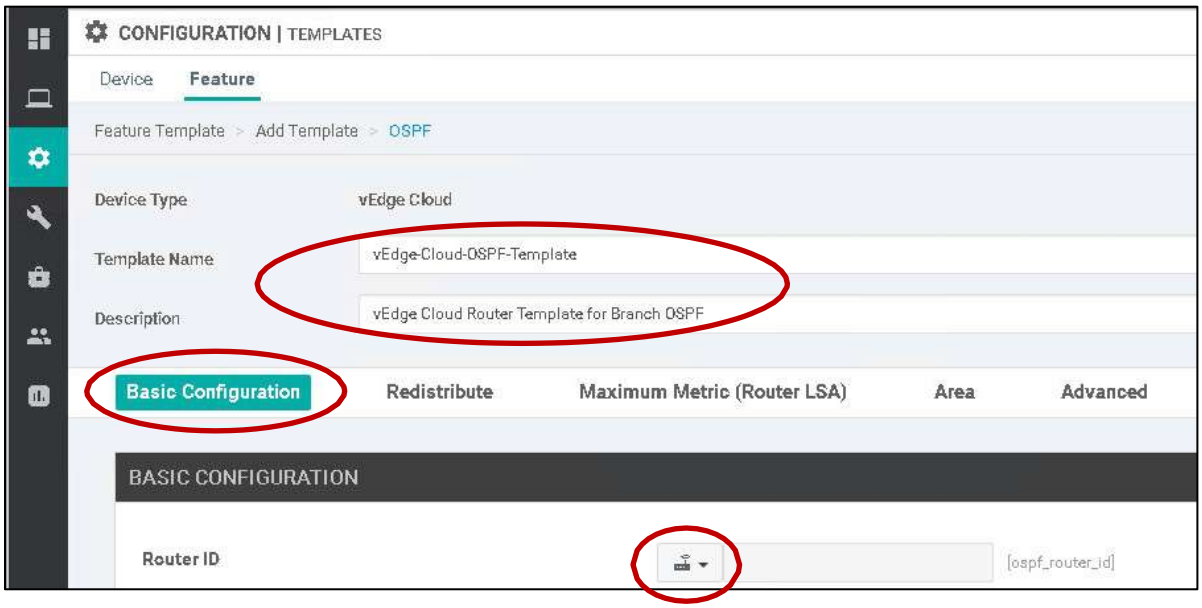
Click **Save** when done.

Step 31 Go to Configuration > Templates > Feature and select +Add Template.

Select **vEdge Cloud** and then on the Template matrix click **OSPF**.

For Template Name enter **"vEdge-Cloud-OSPF-Template"** and enter a **Description**. Select **Device Specific** for the **Router ID** parameter.

Go to the next step.

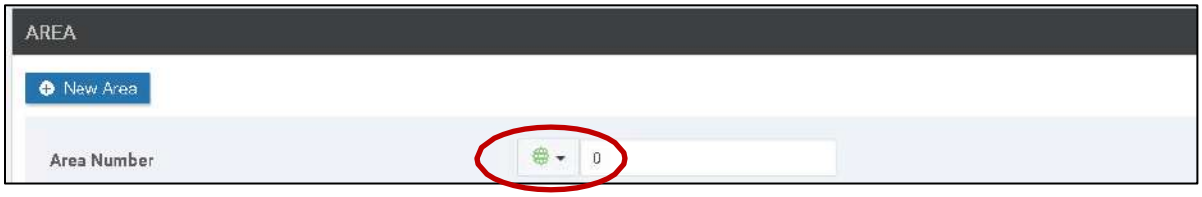


Step 32 Click on **Area** from the categories menu.

Select **+ New Area** to create area 0.

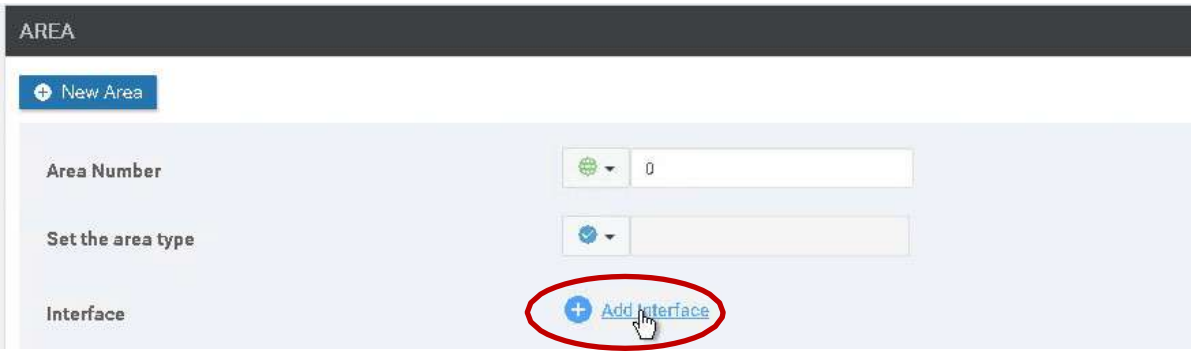
Go to the next step.

Step 33 Use **Global** for the **Area Number** parameter, and then enter the value **"0"**.

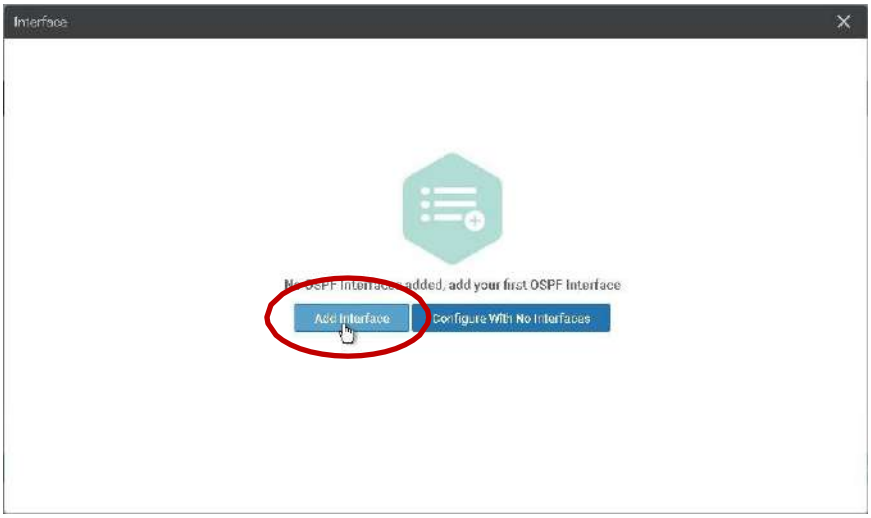


 © 2020 NterOne Corporation

Step 34 Click **+ Add Interface** from the **Area** window.

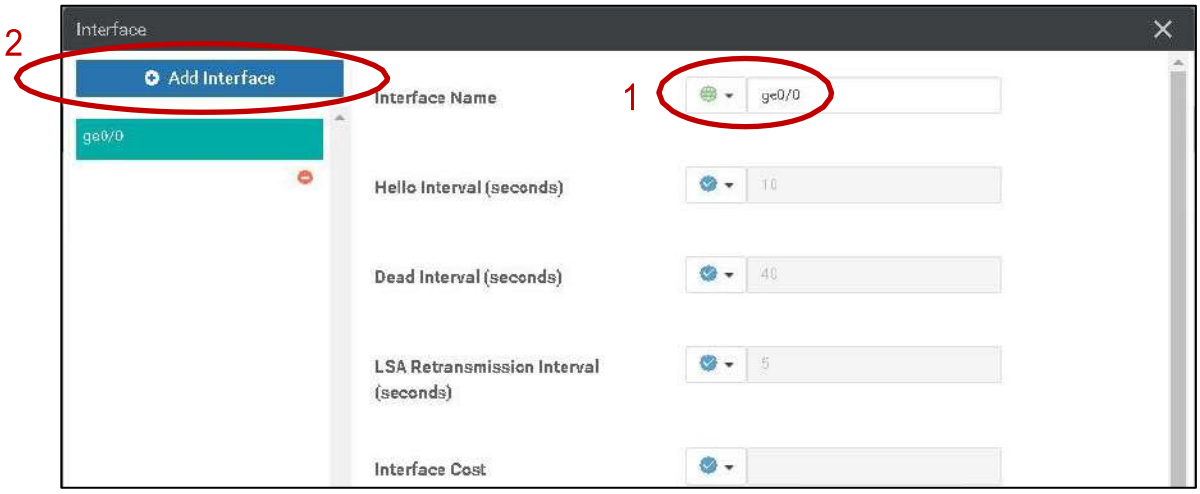


Step 35 Click Add Interface.



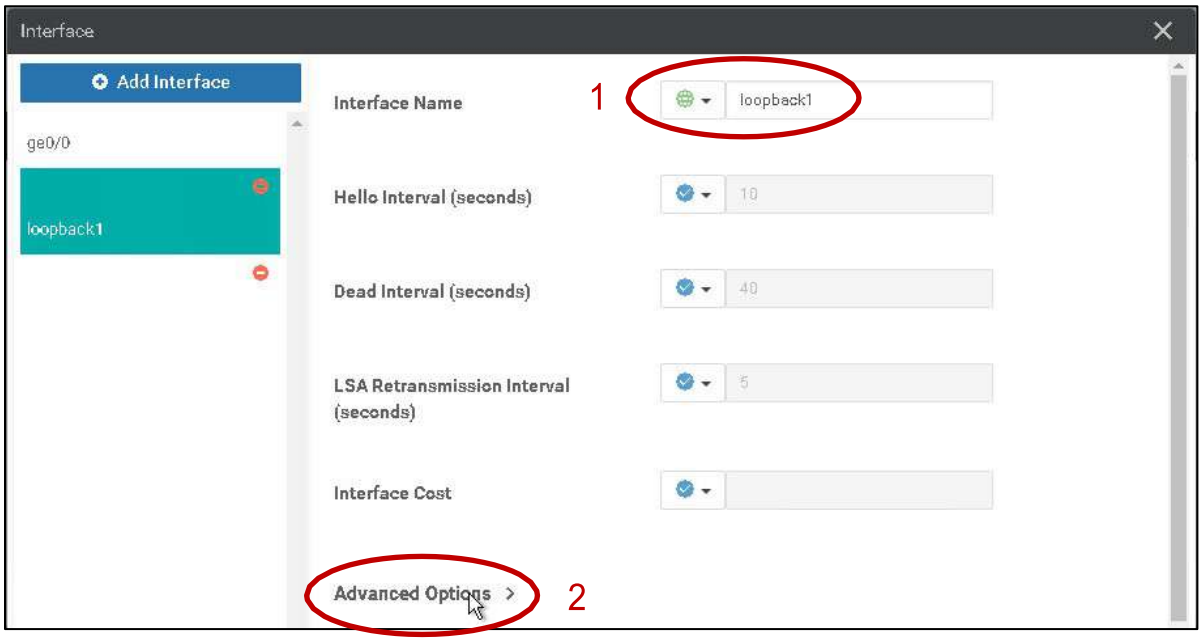
Step 36 Use **Global** for the **Interface Name** parameter and then select/enter **"ge0/0".**

Click **+ Add Interface** to add a **Second** interface to the area.



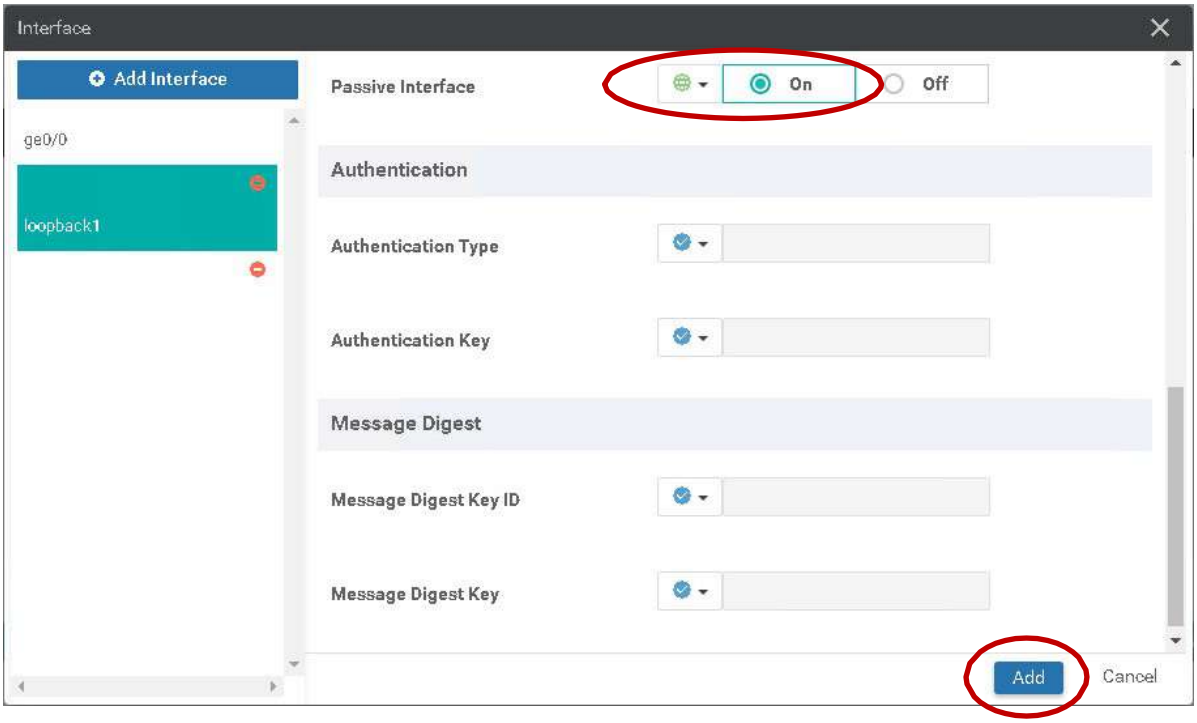
Step 37 Use **Global** for the **Interface Name** parameter, and in the drop-down select/enter **"loopback1"**.

Scroll down and click **Advanced Options**.



Step 38 Select **Global** for the **Passive Interface** parameter, and then click **On**.

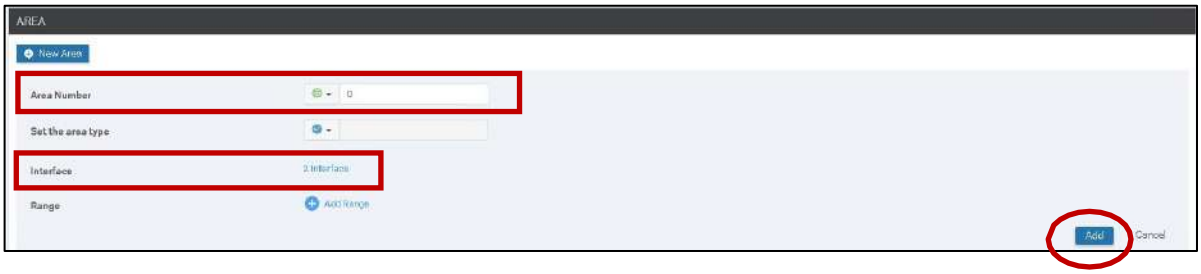
Click **Add** when done.



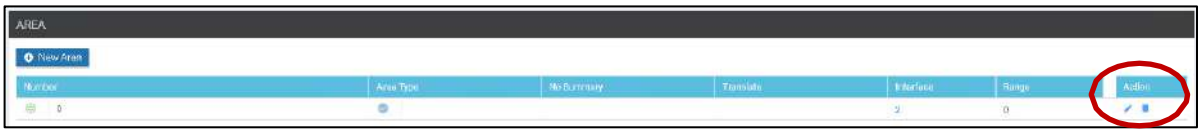
 © 2020 NterOne Corporation

Step 39 Verify that Area Number is **"0"** and for Interface, there should be **2 Interfaces.**

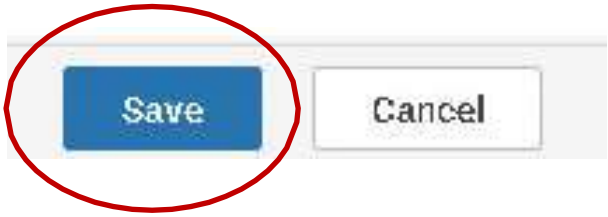
Click **Add**.



Step 40 Area results are displayed. If changes are needed, the **pencil icon** located to the right can be used to edit.



Step 41 Click **Save** to save the OSPF template.



Step 42 Create two more Feature Templates for **SNMP** and **Banner (Optional)**

Step 43 Go to Configuration > Templates > Feature and select +Add Template.

Select **vEdge Cloud** and then on the Template matrix click **SNMP**.

For Template Name enter **"vEdge-Cloud-SNMP-Template"** and enter a **Description**. **Choose or Enter** the following values **in the order listed**.

Shutdown:	(Global) – Select **No**
Name of Devices for SNMP:	(Global) - Enter "**vEdge Cloud Router"**
Contact Person:	(Global) - Enter **"Your Name"** Location
of Device:	(Global) **-** Enter **"San Jose"**

(Click VIEW on SNMP VERSION - if not already selected)

View:	(Click + New View)

Name:	(Global) - Enter “Remote Branch”
Click **Add**	

(Click COMMUNITY)

Community Name:	**(Click + New Community)**
Community:	(Global) - Enter “**public**”
Authorization:	(Global) - Choose “**read-only**”
View:	(Global) - Choose “**Remote Branch**”
Click **Add**.	

Click Save when done.

Step 44 Go to **Configuration > Templates > Feature and select +Add Template.**

Select **vEdge Cloud** and then on the Template matrix click **Banner**.

For Template Name enter **“vEdge-Cloud-Banner-Template”** and enter a **Description**.

Login Banner: (Global) - Enter **“Only authorized individuals are allowed access”**

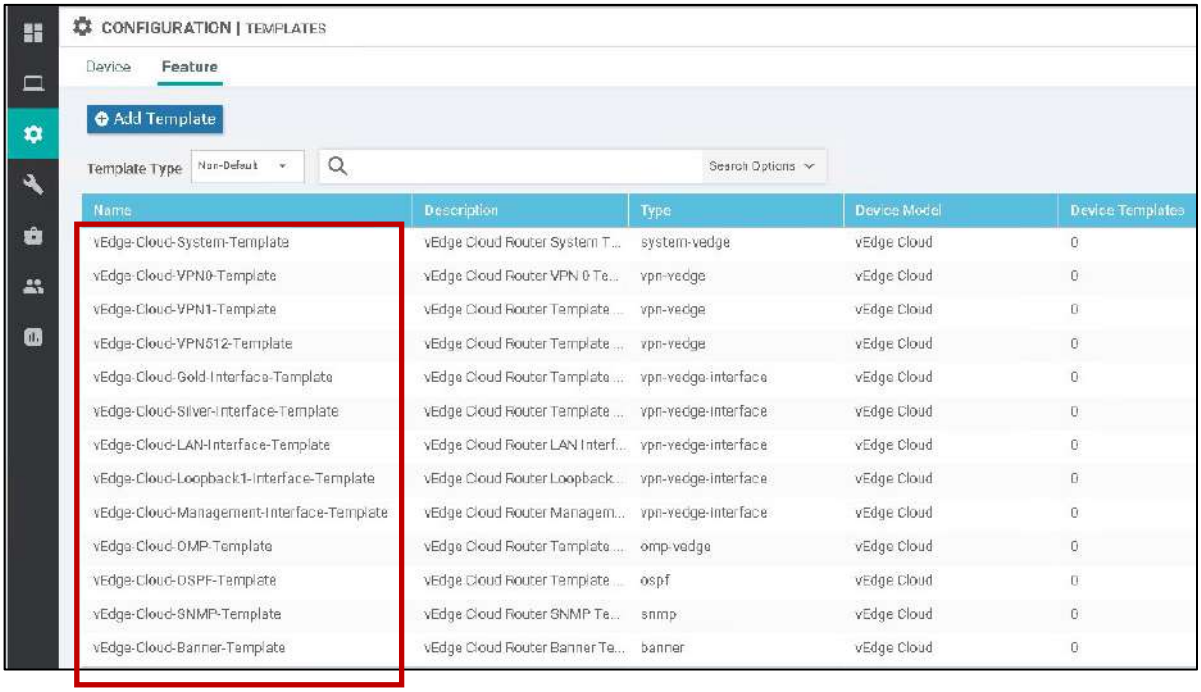
MOTD Banner: (Global) - Enter **“Hello”**

Click **Save** when done.

Step 45 Go to **Configuration > Templates > Feature** and view the list of Feature Templates. There should be a total of 11-13 Templates. If there are any missing determine which ones, and then go back to create them.

Your results should look similar to the following output.

 © 2020 NterOne Corporation

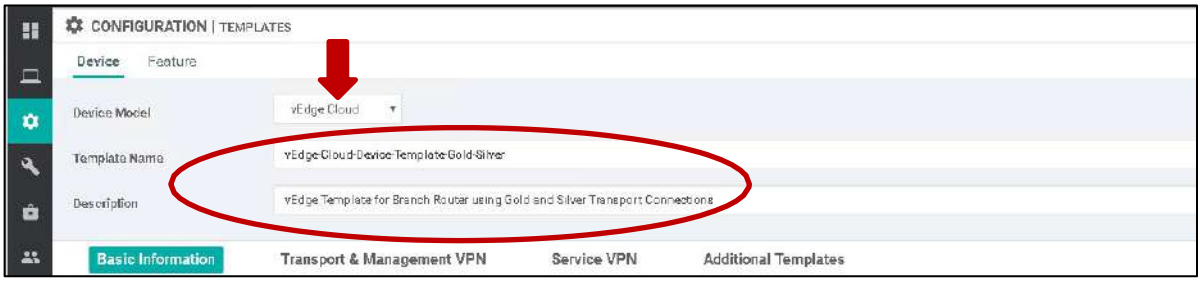


Step 46 **<u>A Device Template will be created next and will be assigned to your Pod's Branch-vE1 router.</u> All of the Feature Templates created in previous steps will be assigned to your Device Template.**

Step 47 Go to **Configuration > Templates > Device and select a +Create Template > From Feature Template**.

Step 48 For **Device Model** select **vEdge Cloud**.

For Template Name enter **"vEdge-Cloud-Device-Template-Gold-Silver"** and a **Description**.



Step 49 There are **Four** categories for a **Device Template**. The next steps outline each category.

Step 50 **Basic Information:** Select the Feature Templates for **System and OMP** created earlier.

Step 51 **Transport & Management VPN**: For VPN 0 and VPN512, select Templates "vEdge-Cloud-VPN0- Template" and "vEdge-Cloud-VPN512-Template" respectively.

Select the "**vEdge-Cloud-Gold-Interface-Template**" for the VPN 0 Interface.

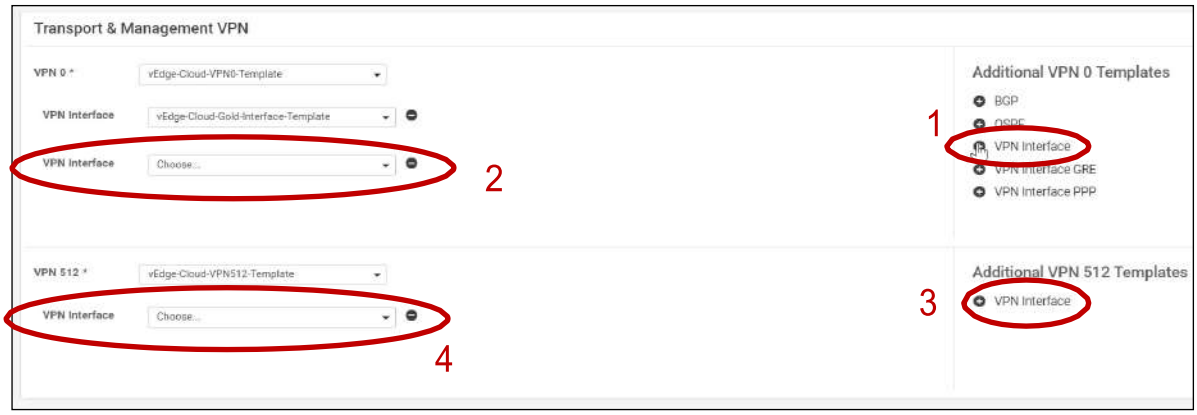
Go to the next step.

Step 52 **Two additional VPN interfaces will be added/created.**

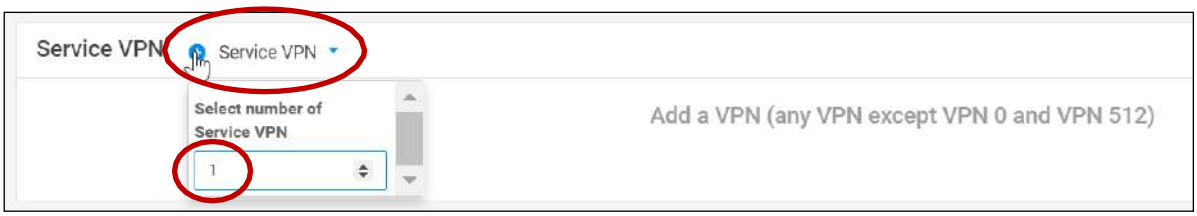
In VPN 0 click on **VPN Interface** located over to the right side, and then select **Vedge-Cloud-Silver- Interface-Template** for the second VPN 0 interface.

Repeat the same procedure for **VPN 512** and select **vEdge-Cloud-Management-Interface-Template** as the only Interface. See below.

 © 2020 NterOne Corporation



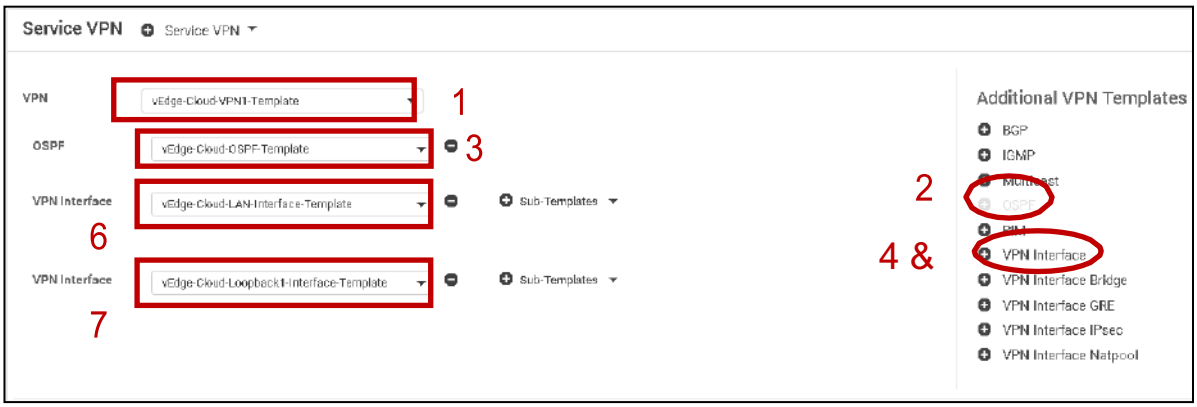
Step 53 <u>**Service VPN:**</u> **HOVER** your mouse over **Service VPN,** and for **Select number of Service VPN,** enter the value **"1"** (if not already entered)**,** and then click the **plus sign +.**



Step 54 For the **VPN** parameter, select **vEdge-Cloud-VPN1-Template** from the drop-down list.

Over to the right side, Click the **plus sign +** on **OSPF** and then select **"vEdge-Cloud-OSPF-Template"**

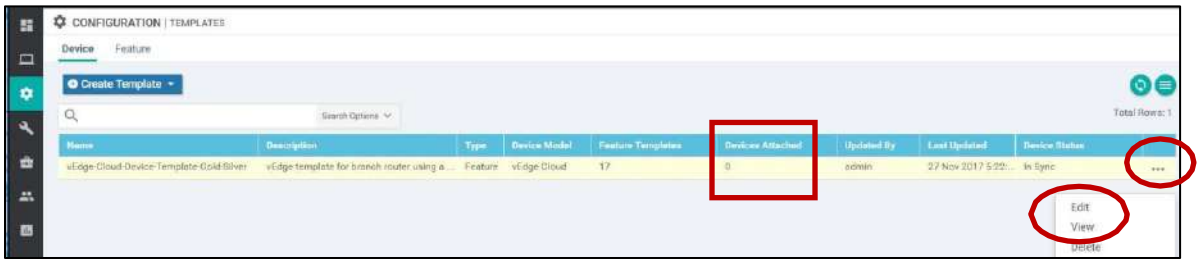
Click the **plus sign +** on VPN Interface and select "vEdge-Cloud-LAN-Interface-Template" **Add another VPN Interface** and select "**vEdge-Cloud-Loopback1-Interface**".



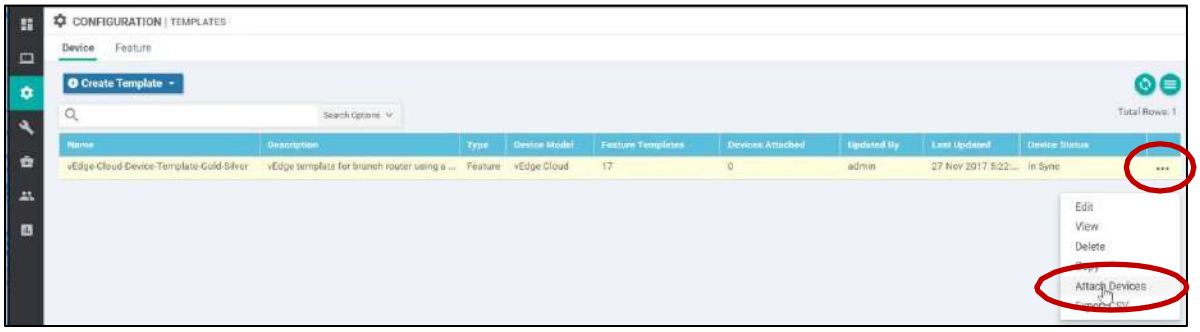
Step 55 <u>**Additional Templates:**</u> **Select** "**vEdge-Cloud-Banner-Template**" and "**vEdge-Cloud-Snmp-Template**" respectively.

Step 56 Click **Create** when done. You will be redirected to the **Templates > Device** List.

Step 57 Verify the **Device Template** entry. Select **…** over on the right to view and modify if necessary. Notice the Template has not been attached to any device as of yet.

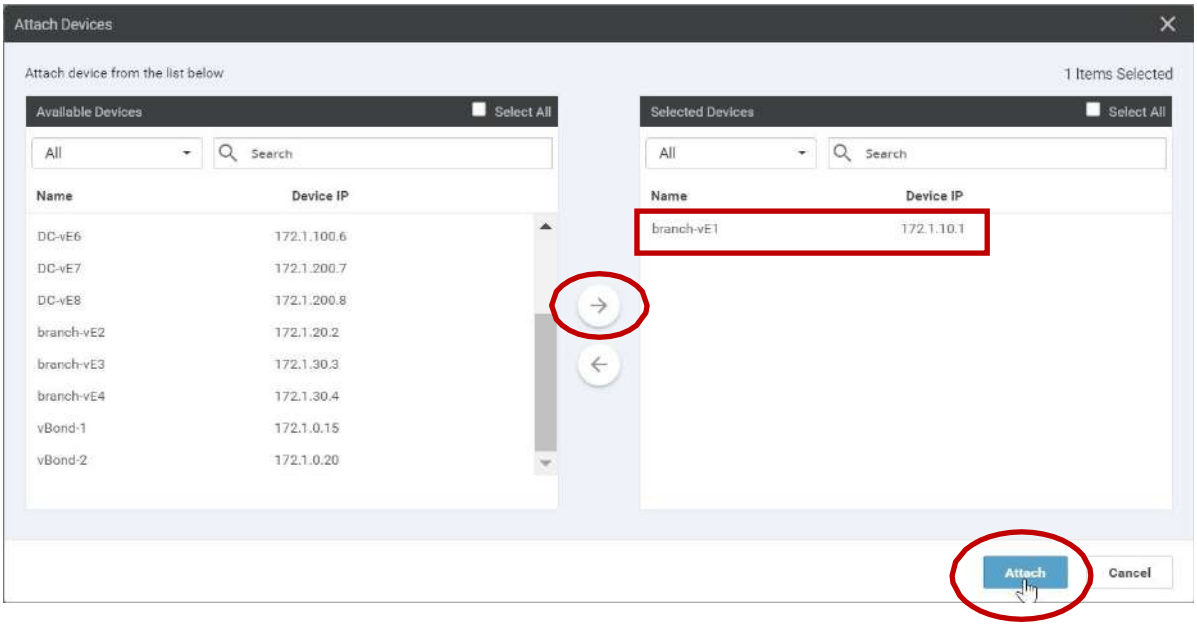


Step 58 Select the actions menu **…** again and on the drop-down list select **Attach Devices**.



 © 2020 NterOne Corporation

Step 59 The **Attach Devices** window will appear. Select the **branch-vE1** vEdge router from the **Available Devices** list and add it to the right under **Selected Devices**. Click **Attach** when done.



Step 60 If not already redirected, then go to **Configuration > Templates > Device**.

Click on **…** and select **Edit Device Template**.

You are required to enter Device Specific Values.

Step 61 Fill in for the **Device Specific** Values. CORRECTLY identify the IPv4 Interface by **hovering the mouse** over each of the **IPv4 Address(vpn_if_ipv4_address)** parameters and then pan over to the right, and enter the **correct IP Address** (Look below).

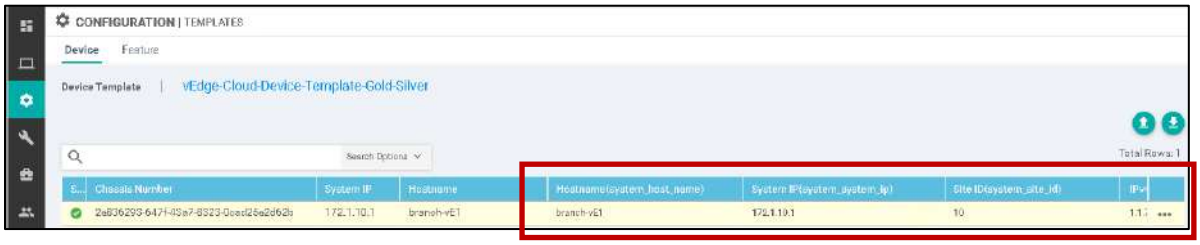
Note **The parameters may not be listed in the same order**. Your results should be similar to the following:

System IP	**172.1.10.1**
Hostname(system_host_name)	**branch-vE1**
System IP(system_system_ip)	**172.1.10.1**
Site ID(system_site_id)	**10**

IPv4 Address(vpn_if_ipv4_address) ge0/1	**1.1.7.11/24**
IPv4 Address(vpn_if_ipv4_address) ge0/2	**1.2.9.11/24**
Router ID(ospf_router_id)	**172.1.10.1**
IPv4 Address(vpn_if_ipv4_address) ge0/0	**10.2.50.11/24**
IPv4 Address(vpn_if_ipv4_address) loopback1	**99.99.99.99/32**

Step 62 Click **Update** when value entry has been completed.

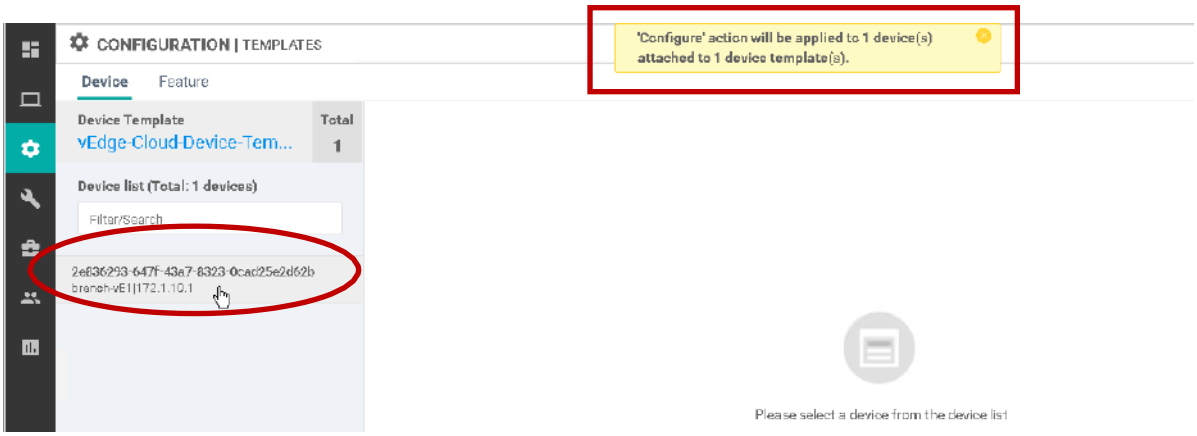
Step 63 You'll be redirected back to the **Device Template** window.



Step 64 Click Next.

 © 2020 NterOne Corporation

Step 65 On the next screen select your vEdge router from the **Device list**.



Step 66 Select the vEdge device on the left, then the configuration to be **pushed** will be displayed over to the right.

Scroll up or down to verify what settings will be **pushed** to the device.

If there are any **missing features or incorrect values** click **Cancel** and go back to the appropriate step and add or modify respectively. Repeat the previous steps to attach the Template if necessary.

Verify with the following configuration excerpt.

```
viptela-system:system
device-model            vedge-cloud
host-name               branch-vE1

system-ip               172.1.10.1
domain-id               1
site-id                 10
controller-group-list   10
admin-tech-on-failure
no route-consistency-check
sp-organization-name    "NterOne Lab"
organization-name       "NterOne Lab"
console-baud-rate       115200
```

```
     vbond vbond.demo.local port 12346
     aaa
      auth-order local radius tacacs
      usergroup basic
       task system read write
       task interface read write
      !
      usergroup netadmin
      !
      usergroup operator
       task system read
       task interface read
       task policy read
       task routing read
       task security read
      !
      user admin
       password
$6$siwKBQ==$wT2lUa9BSreDPI6gB8sl4E6PAJoVXgMbgv/whJ8F1C6sWdRazdxorYYTLrL6
syiG6qnLABTnrE96HJiKF6QRq1
      !
     !
     logging
      disk
       enable
      !
     !
    !
    omp
     no shutdown
     graceful-restart
     advertise ospf external
     advertise connected
     advertise static
    !
    snmp
     no shutdown
     contact  Mike
     name     "vEdge Cloud Router"
     location "San Jose"
     view "Remote Branch"
```

 © 2020 NterOne Corporation

```
 !
 community public
  view          "Remote Branch"
  authorization read-only
 !
!
security
 ipsec
  authentication-type sha1-hmac ah-sha1-hmac
 !
!
banner
 login "Only authorized Individuals are allowed access"
 motd  Hello
!
vpn 0
 dns 1.4.28.89 primary
 interface ge0/1
  ip address 1.1.7.11/24
  tunnel-interface
   encapsulation ipsec
   color gold
   no allow-service bgp
   allow-service dhcp
   allow-service dns
   allow-service icmp
   no allow-service sshd
   no allow-service netconf
   no allow-service ntp
   no allow-service ospf
   no allow-service stun
   allow-service https
  !
  no shutdown
 !
 interface ge0/2
  ip address 1.2.9.11/24
  tunnel-interface
```

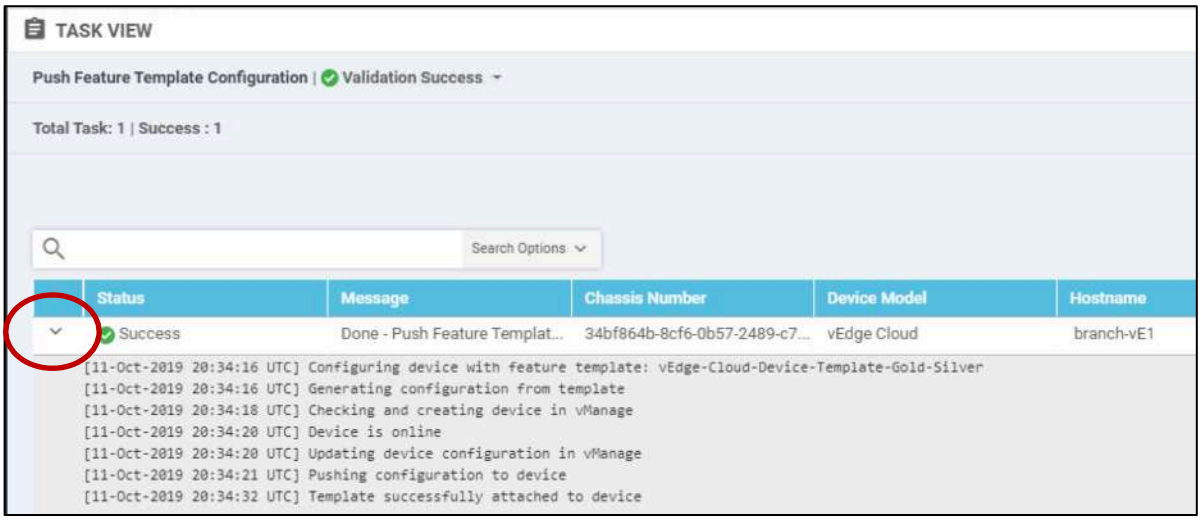
```
    encapsulation ipsec
    color silver
    no allow-service bgp
    allow-service dhcp
    allow-service dns
    allow-service icmp
    no allow-service sshd
    no allow-service netconf
    no allow-service ntp
    no allow-service ospf
    no allow-service stun
    allow-service https
   !
   no shutdown
  !
  ip route 0.0.0.0/0 1.1.7.22
  ip route 0.0.0.0/0 1.2.9.23
!
vpn 1
 router
  ospf
   router-id 172.1.10.1
   timers spf 200 1000 10000
   area 0
    interface ge0/0
    exit
    interface loopback1
     passive-interface
    exit
   exit
  !
 !
 interface ge0/0
  ip address 10.2.50.11/24
  no shutdown
 !
 interface loopback1
  ip address 99.99.99.99/32
  no shutdown
 !
```

 © 2020 NterOne Corporation

```
 !
 vpn 512
  interface eth0
   ip address 10.82.83.111/20

 no shutdown
  !
 !
!
```

Step 67 Click **Configure Devices** to proceed with the Device Template attachment.

Step 68 You're redirected to **Task view page** to monitor the progress by selecting the right arrow . Confirm that it was successful.



Step 69 If the **Status** is **Failed** go back and correct whatever necessary or **contact your instructor for assistance**.

Step 70 Go to **Configuration > Devices** and verify the vEdge router and its associated Device Template.

The Device Mode has now been changed **to vManage mode from CLI mode** and the Template has been assigned. When the device is in vManage Mode you cannot make any changes in the Device CLI.

branch-vE2	172.1.20.2	20	vManage	vedge6
branch-vE1	172.1.10.1	10	vManage	vEdge-Cloud-Device-Template-Gold-Silver

Step 71 **Login to your vEdge router** and verify that the Overlay Network is functioning once again.

Step 72 Use the following commands to confirm vEdge operation.

```
branch-vE1# show omp peers                    (Output may vary)
R -> routes received
I -> routes installed
S -> routes sent
                        DOMAIN    OVERLAY   SITE
PEER            TYPE    ID        ID        ID        STATE    UPTIME          R/I/S
-----
172.1.0.16      vsmart  1         1         1011      up       0:01:31:57      20/10/4
172.1.0.19      vsmart  1         1         1013      up       0:01:31:55      26/7/4

branch-vE1# show control connections     (Formatted for clarity)(Output may vary)
```

 © 2020 NterOne Corporation

```
                                                    PEER                    PEER
PEER    PEER PEER          SITE  DOMAIN  PEER       PRIV PEER                PUB   LOCAL   STATE
TYPE    PROT SYSTEM IP     ID    ID      PRIVATE IP PORT PUBLIC      IP       PORT  COLOR
-----------------------------------------------------------------------------------------------
vsmart  dtls 172.1.0.16    1011  1       1.3.26.26  1244 1.3.26.26           12446 gold    up
                                                     6
vsmart  dtls 172.1.0.16    1011  1       1.3.26.26  1244 1.3.26.26           12446 silver  up
                                                     6
vsmart  dtls 172.1.0.17    1012  1       1.4.27.27  1244 1.4.27.27           12446 gold    up
                                                     6
vsmart  dtls 172.1.0.17    1012  1       1.4.27.27  1244 1.4.27.27           12446 silver  up
                                                     6
vmanage dtls 172.1.0.18    1001  0       1.4.28.28  1284 1.4.28.28           12846 gold    up
                                                     6
```

branch-vE1# **show ip routes** *(Formatted for clarity)(Output may vary)*

```
Codes Proto-sub-type:
     IA -> ospf-intra-area, IE -> ospf-inter-area,
     E1 -> ospf-external1, E2 -> ospf-external2,
     N1 -> ospf-nssa-external1, N2 -> ospf-nssa-external2,
     e -> bgp-external, i -> bgp-internal


Codes Status flags:
     F -> fib, S -> selected, I -> inactive,
     B -> blackhole, R -> recursive


                                          PROTOCOL NEXTHOP NEXTHOP NEXTHOP
VPN              PREFIX          PROTOCOL  SUB TYPE IF NAME ADDR    VPN TLOC IP
                 COLOR           ENCAP STATUS
-------------------------------------------------------------------------------

 0      0.0.0.0/0       static       - ge0/1   1.1.7.22  -  -          -      -      F,S
 0      0.0.0.0/0       static       - ge0/2   1.2.9.23  -  -          -      -      F,S
 0      1.1.7.0/24      connected    - ge0/1   -         -  -          -      -      F,S
 0      1.2.9.0/24      connected    - ge0/2   -         -  -          -      -      F,S
 0      172.1.10.1/32   connected    - system  -         -  -          -      -      F,S
 1      0.0.0.0/0       omp          - -       -         -            gold   ipsec  F,S
                                                             172.1.100.
                                                           5
 1      0.0.0.0/0       omp          - -       -         -            gold   ipsec  F,S
                                                             172.1.100.
                                                           6
 1      0.0.0.0/0       omp          - -       -         -            silve  ipsec  F,S
                                                             172.1.200. r
                                                           7
 1      0.0.0.0/0       omp          - -       -         -            silve  ipsec  F,S
                                                             172.1.200. r
                                                           8
 1      10.2.50.0/24    ospf        IA ge0/0   -         -  -          -      -      -
 1      10.2.50.0/24    connected    - ge0/0   -         -  -          -      -      F,S
 1      10.2.51.0/24    omp          - -       -         -  172.1.20.2 gold   ipsec  F,S
 1      10.2.52.0/24    omp          - -       -         -  172.1.30.3 gold   ipsec  F,S
```

```
1      10.2.53.0/24     omp          - -       -         -  172.1.30.4 gold   ipsec  F,S
1      10.2.54.0/24     omp          - -       -         -             gold   ipsec  F,S
                                                             172.1.100.
                                                           5
1      10.2.55.0/24     omp          - -       -         -             gold   ipsec  F,S
                                                             172.1.100.
                                                           6
1      10.2.56.0/24     omp          - -       -         -             silve  ipsec  F,S
                                                             172.1.200. r
                                                           7
1      10.2.57.0/24     omp          - -       -         -             silve  ipsec  F,S
                                                             172.1.200. r
                                                           8
1      10.2.58.0/24     omp          - -       -         -             gold   ipsec  F,S
                                                             172.1.100.
                                                           5
1      10.2.58.0/24     omp          - -       -         -             gold   ipsec  F,S
                                                             172.1.100.
                                                           6
1      10.2.58.0/24     omp          - -       -         -             silve  ipsec  F,S
                                                             172.1.200. r
                                                           7
1      10.2.58.0/24     omp          - -       -         -             silve  ipsec  F,S
                                                             172.1.200. r
                                                           8
1      39.39.39.39/32   omp          - -       -         -             gold   ipsec  F,S
                                                             172.1.100.
                                                           5
1      39.39.39.39/32   omp          - -       -         -             gold   ipsec  F,S
                                                             172.1.100.
                                                           6
1      99.99.99.99/32   ospf   IA loopback1- -      -      -      -      - connected -
1      99.99.99.99/32          loopback1-   -      -      -      -      F,S
```

--- VPN 512 OMITTED ---

```
branch-vE1# show ipsec local-sa                (Formatted for clarity)(Output may
                                                                vary)
                                   SOURCE          SOURCE            SOURCE
TLOC ADDRESS  TLOC COLOR   SPI     IPv4            IPv6              PORT    KEY HASH
-----------------------------------------------------------------------------------
-
172.1.10.1    gold         267     1.1.7.11        ::                12346   *****0eb5
172.1.10.1    gold         268     1.1.7.11        ::                12346   *****0eb5
172.1.10.1    silver       267     1.2.9.11        ::                12346   *****2d3f
172.1.10.1    silver       268     1.2.9.11        ::                12346   *****2d3f
```

Activity Verification

You have completed this task when you attain these results:

- You have successfully created Feature Templates using vManage
- You have successfully created a Device Template using vManage
- You have successfully attached a Device Template to a vEdge router using vManage

 © 2020 NterOne Corporation

Task 2: Deploy CLI Based Device Template

In this Task you will deploy a Device Template using a CLI Template.

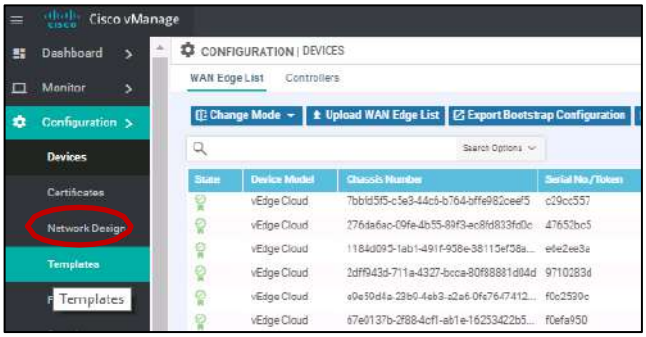
Activity Procedure

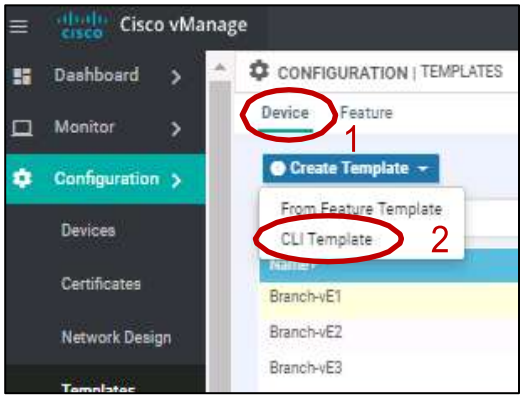
Complete these steps:

Step 1 Log in to vManage. You will use vManage to create Templates.

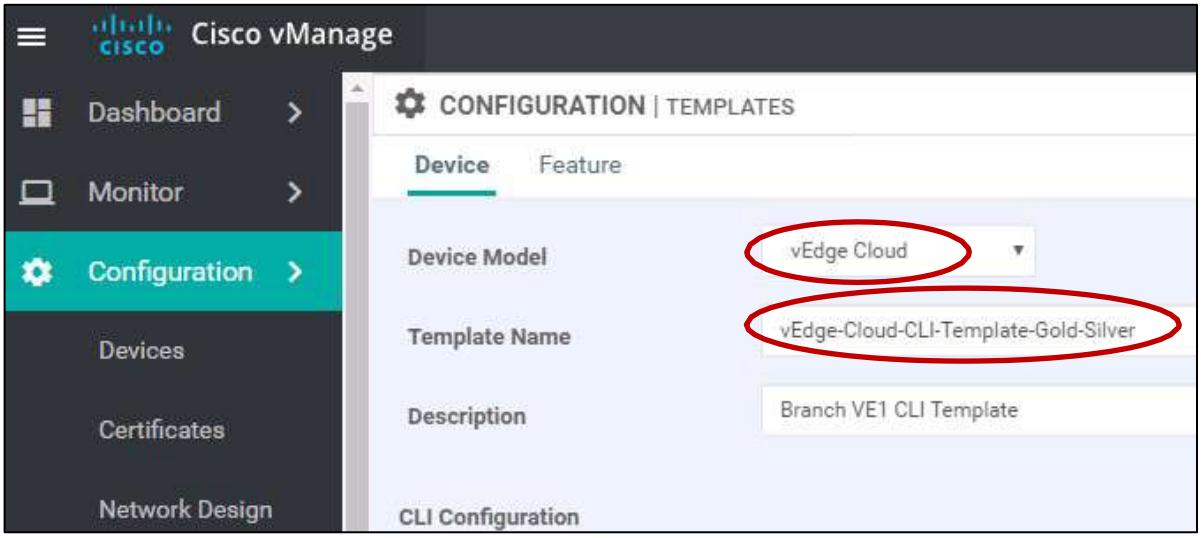
Step 2 A Device **CLI Template** will be created.

Step 3 Go to **Configuration > Templates** and select **Device,** and then click **+ Create Template > CLI Template** from the top left corner of the templates window.

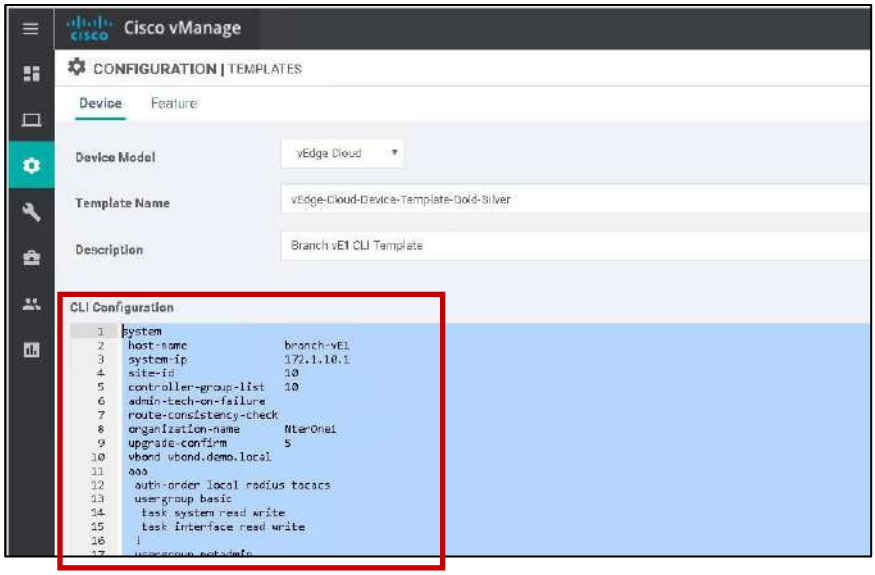




Step 4 Select **vEdge Cloud** for the Device Model. For the Template Name use "vEdge-Cloud-CLI-Template- Gold-Silver" and then provide a Description.

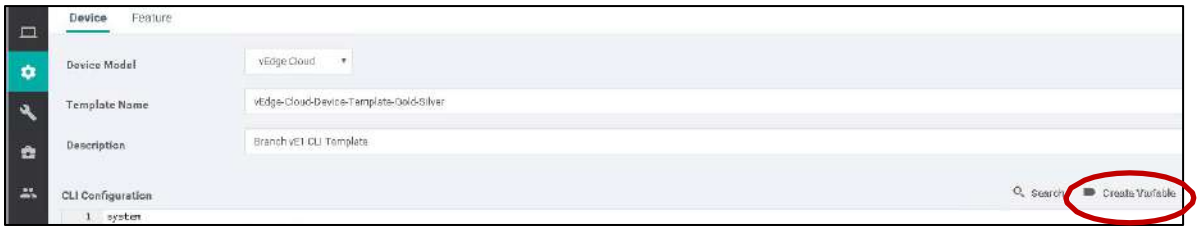


Step 5 Paste the contents of the file **"Branch-vE1.txt"** (SDWAN Lab Files folder on Admin PC desktop) intothe **CLI Configuration** area of the window. (SDWAN Lab Files folder on the desktop) into the **CLI Configuration** area of the window. **Make sure you edit the IP address of VPN 512 if necessary**. **Do not** click add just yet as variables need to be created.



Step 6 The **CLI Template will be populated with variables**. The variables will be used as placeholders for **Device Specific values** when applying the template to multiple devices.

Step 7 Over to the right side of the window click **Create Variable**.



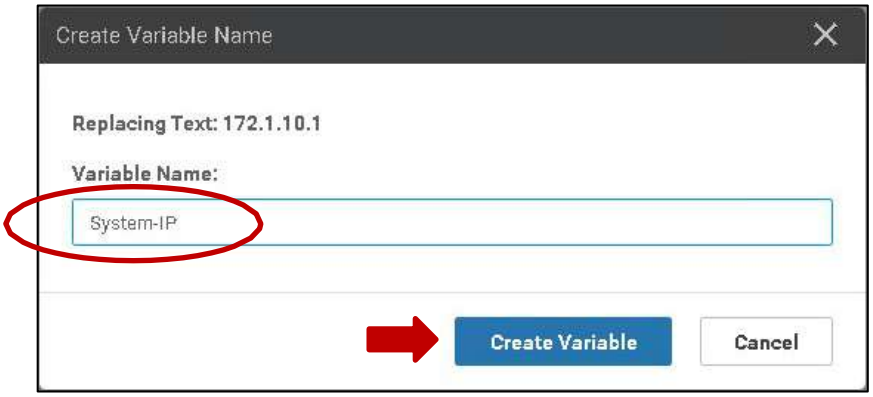
Step 8 A window will appear that indicates text needs to be **highlighted first** before creating a variable. Click Cancel.

Step 9 **First, Highlight the system-ip** value 172.1.10.1

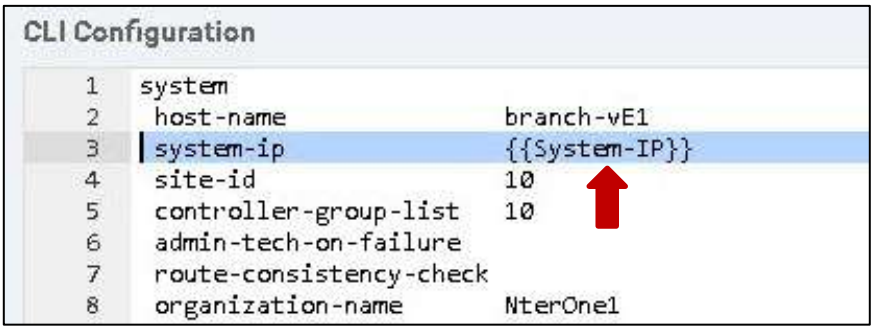
Step 10 Over to the right side of the window once again select **Create Variable**.

 © 2020 NterOne Corporation

Step 11 For the Variable Name use "System-IP" and then click Create Variable.



Step 12 The value for the **system-ip parameter** should now contain the variable name **{{System-IP}}** instead of the value itself.



Step 13 Repeat Variable creation for the following parameters. Use the variable names shown below.

Parameter	Variable Name
host-name	**Host-Name**
site-id	**Site-ID**
controller-group-list	**Controller-Group**
router-id	**System-IP** *(**Note:** "System-IP" variable is used twice)*
int ge0/1 – ip address	**Int-ge0_1-ip-address**
int ge0/2 – ip address	**Int-ge0_2-ip-address**
int ge0/0 – ip address	**Int-ge0_0-ip-address**
int loopback1 – ip address	**Int-loopback1-ip-address**
1.1.7.22	**Int-ge0_1-GW** *(**Note:** Highlight only the GW value itself)*
1.2.9.23	**Int-ge0_2-GW** *(**Note:** Highlight only the GW value itself)*
int eth0 – ip address	**Int-eth0-ip-address**

Step 14 **Verify your results** with the following code excerpt.

```
system
  host-name               {{Host-Name}}
  system-ip               {{System-IP}}
  site-id                 {{Site-ID}}
  controller-group-list   {{Controller-Group}}
  admin-tech-on-failure
  no route-consistency-check
  sp-organization-name    "NterOne Lab"
  organization-name       "NterOne Lab"
  vbond vbond.demo.local
  aaa
   auth-order local radius tacacs
   usergroup basic
    task system read write
    task interface read write
   !
   usergroup netadmin
   !
   usergroup operator
    task system read
    task interface read
    task policy read
    task routing read
    task security read
   !
   user admin
    password
$6$siwKBQ==$wT2lUa9BSreDPI6gB8sl4E6PAJoVXgMbgv/whJ8F1C6sWdRazdxorYYTL
rL6syiG6qnLABTnrE96HJiKF6QRq1
   !
  !
  logging
   disk
    enable
   !
  !
 !
 omp
  no shutdown
  graceful-restart
```

 © 2020 NterOne Corporation

```
 advertise ospf external
 advertise connected
 advertise static
!
security
 ipsec
  authentication-type sha1-hmac ah-sha1-hmac
 !
!
snmp
 no shutdown
 contact  "Richard Rowe"
 name     "vEdge Cloud Router"
 location "San Jose"
 view "Remote Branch"
 !
 community public
  view          "Remote Branch"
  authorization read-only
 !
!
banner
 login "Only authorized individuals are allowed access"
 motd  "Hello!"
!


vpn 0
 dns 1.4.28.89 primary
 interface ge0/1
  ip address {{Int-ge0_1-ip-address}}
  tunnel-interface
   encapsulation ipsec
   color gold
   no allow-service bgp
   allow-service dhcp
   allow-service dns
   allow-service icmp
   no allow-service sshd
   no allow-service netconf
```

```
    no allow-service ntp
    no allow-service ospf
    no allow-service stun
   !
   no shutdown
  !
  interface ge0/2
   ip address {{Int-ge0_2-ip-address}}
   tunnel-interface
    encapsulation ipsec
    color silver
    no allow-service bgp
    allow-service dhcp
    allow-service dns
    allow-service icmp
    no allow-service sshd
    no allow-service netconf
    no allow-service ntp
    no allow-service ospf
    no allow-service stun
   !
   no shutdown
  !
  ip route 0.0.0.0/0 {{Int-ge0_1-GW}}
  ip route 0.0.0.0/0 {{Int-ge0_2-GW}}
 !
 vpn 1
  router
   ospf
    router-id {{System-IP}}
    timers spf 200 1000 10000
    area 0
     interface ge0/0
     exit
     interface loopback1
      passive-interface
     exit
    exit
   !
  !
```

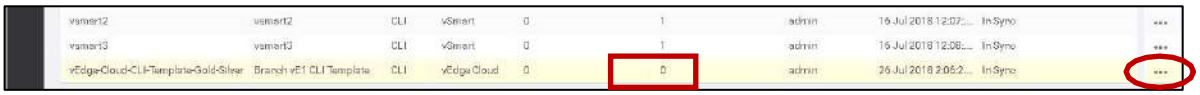
 © 2020 NterOne Corporation

```
 interface ge0/0
  ip address {{Int-ge0_0-ip-address}}
  no shutdown
 !
 interface loopback1
  ip address {{Int-loopback1-ip-address}}
  no shutdown
 !
!
vpn 512
 interface eth0
  ip address {{Int-eth0-ip-address}}
  no shutdown
 !
```

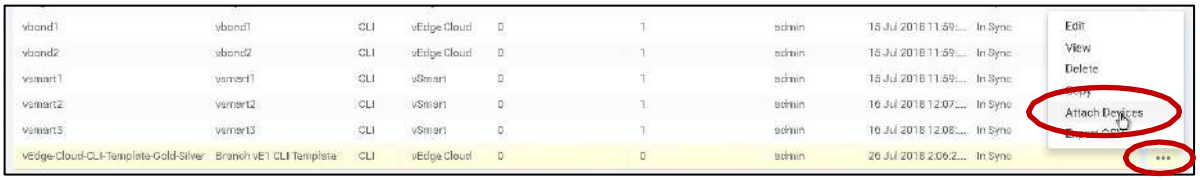
Step 15 Once you're satisfied with the results, click **Add**.

Step 16 Verify the **Device Template** entry. Click on **…** over from the right side to view and edit if necessary.

Notice the template has not been attached to any device as of yet.

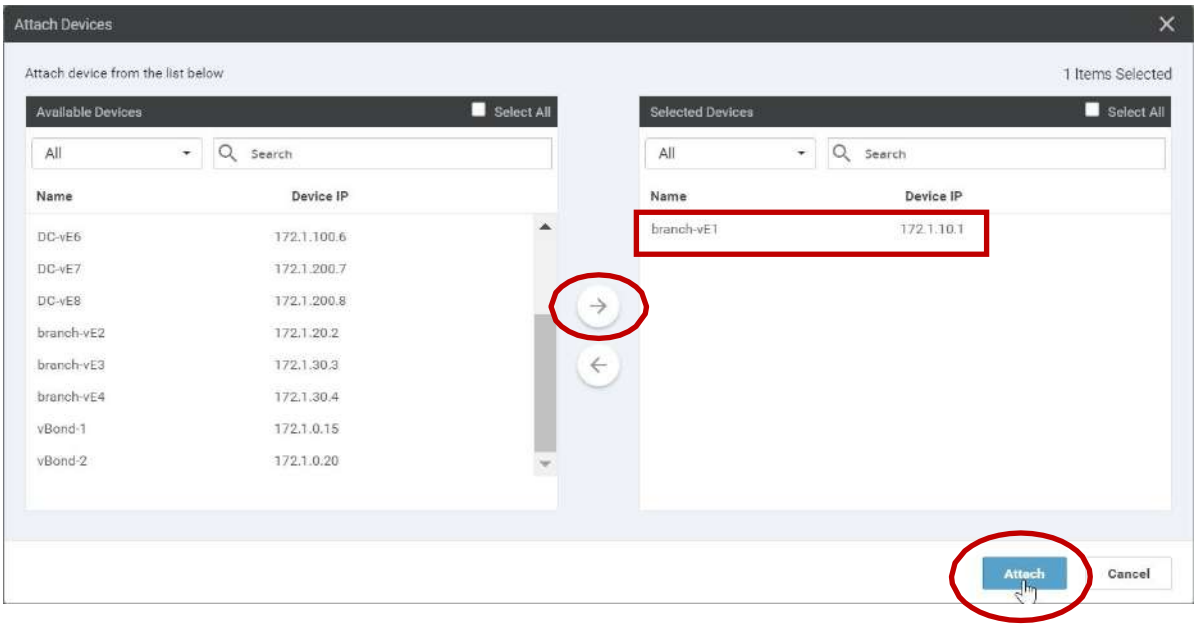


Step 17 Once again click on **…** and from the drop-down menu select **Attach Devices**.

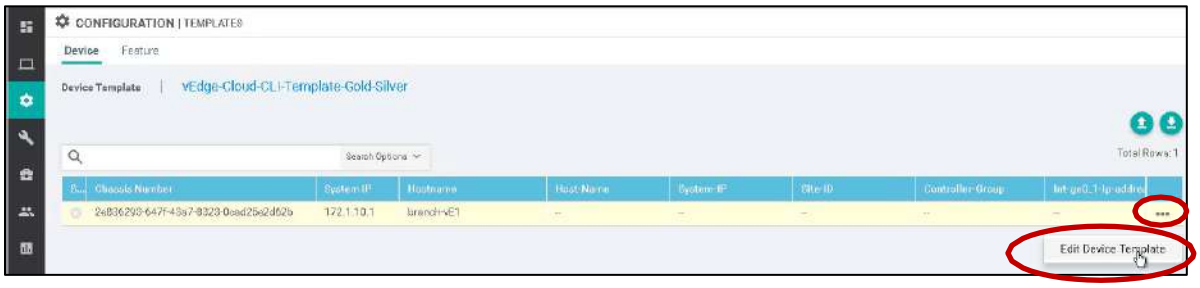


Step 18 The **Attach Devices** window will appear. Select the branch-vE1 router from the **Available Devices** list and add it to the right under **Selected Devices**. Click **Attach** when done.

Note: In previous versions of software, you would first have to detach the router from an existing template, prior to attaching it to a new templete.



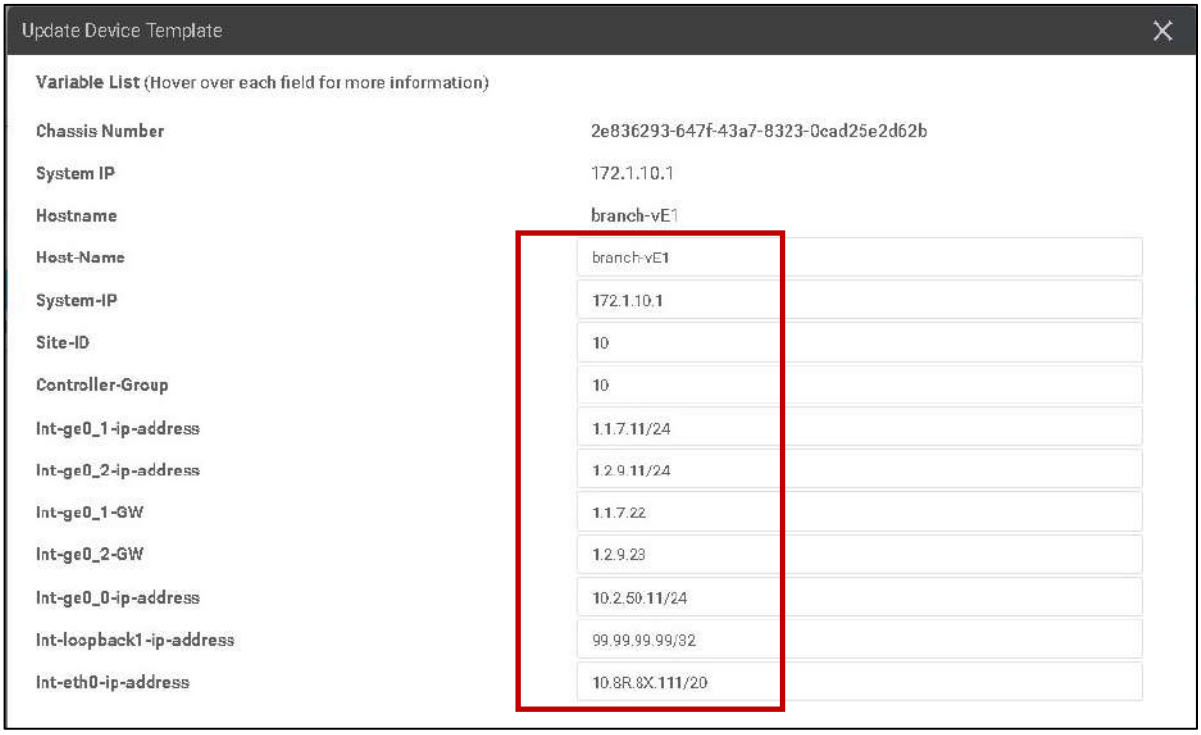
Step 19 Over to the right-side click on **…** and select **Edit Device Template**. You are required to enter **Device Specific Values**.



Step 20 Fill in for the **Device Specific** Values. Use the values shown below.

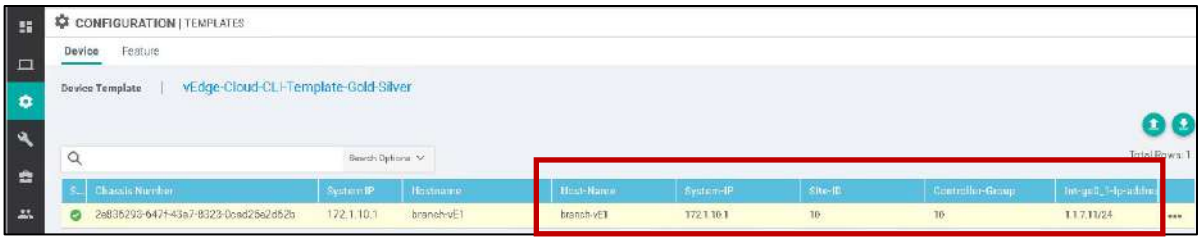
Host-Name	**branch-vE1**
System-IP	**172.1.10.1**
Site-ID	**10**
Controller-Group	**10**
Int-ge0_1-ip-address	**1.1.7.11/24**
Int-ge0_2-ip-address	**1.2.9.11/24**
Int-ge0_1-GW	**1.1.7.22**
Int-ge0_2-GW	**1.2.9.23**
Int-ge0_0-ip-address	**10.2.50.11/24**
Int-loopback1-ip-address	**99.99.99.99/32**
Int-eth0-ip-address	**10.82.83.111/20**

 © 2020 NterOne Corporation



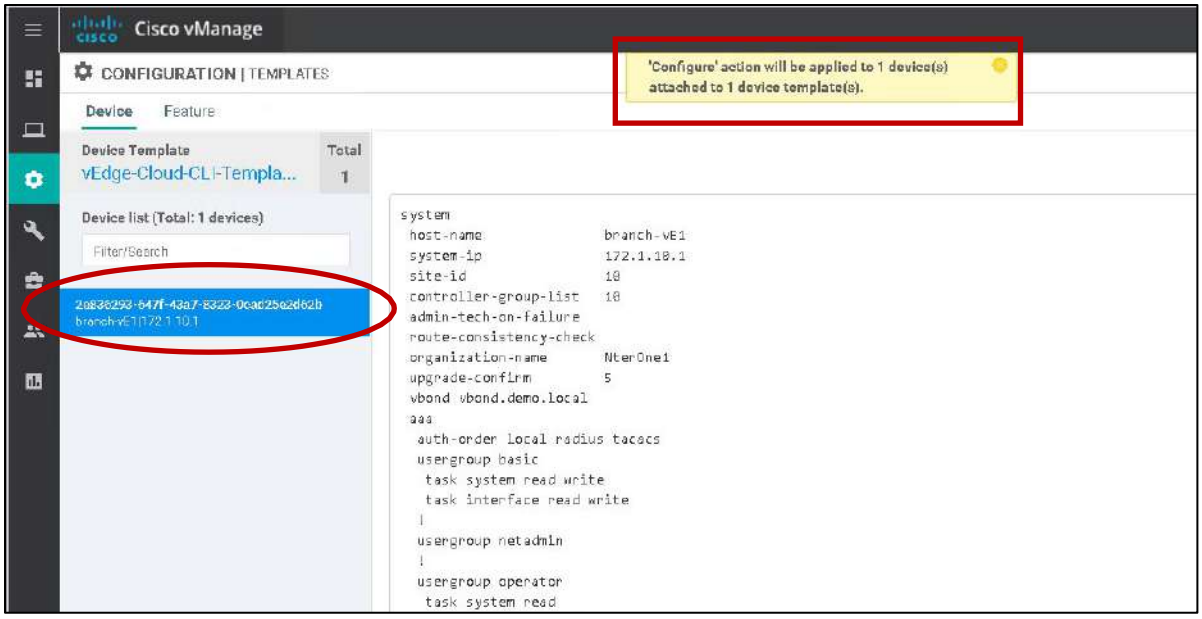
Step 21 Click **Update** when done.

Step 22 You'll be brought back to the **Device Template** window.



Step 23 Click Next.

Step 24 On the next screen select your vEdge router from the **Device list**.



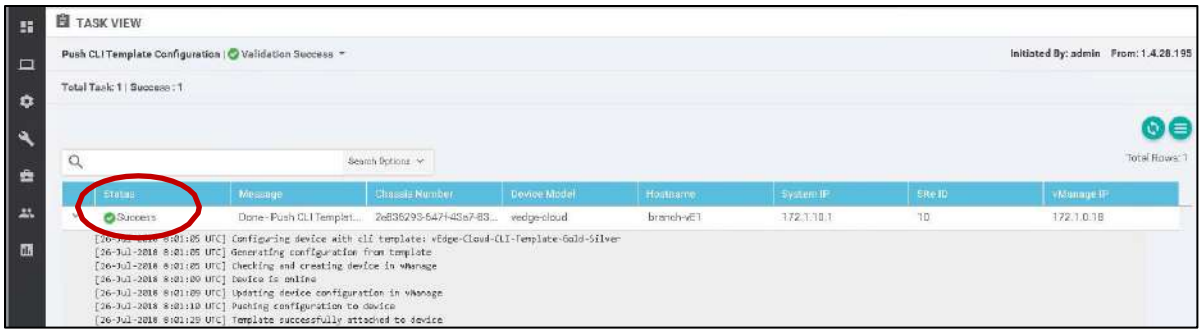
Step 25 Once the vEdge device is selected, then the configuration to be **pushed** will be displayed over on the right.

Scroll up or down to verify what settings will be **pushed** to the device.

If there are any **missing features or incorrect values** click **Cancel** and go back to the appropriate step and add or modify respectively. If necessary, repeat the previous steps to attach the Template.

Step 26 Click **Configure Devices** to proceed with the Device Template attachment.

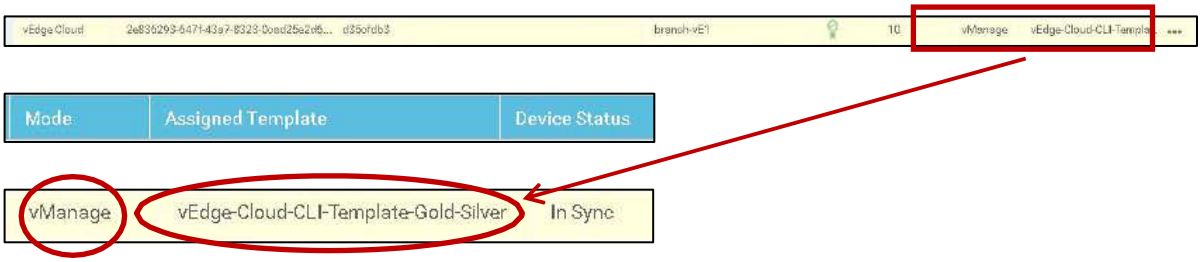
Step 27 You are redirected to **Task View** to monitor the progress. Confirm that it was successful.



Step 28 If the **Status** is **Failed** go back and correct whatever necessary or **contact your instructor for assistance**. A simple typo can cause it to fail. Optionally, you can also re-attach the previous **vEdge-Cloud-Device-Template-Gold-Silver** template to your vEdge router and continue on to Lab 8 if you prefer.

 © 2020 NterOne Corporation

Step 29 Go to **Configuration > Devices** and verify the vEdge router and its associated CLI Template. The Mode has now been changed from **CLI to vManage** and the Template has been assigned. When the device is in vManage Mode you cannot make any changes in the Device CLI.



Step 30 **Login to your vEdge router** and verify that the Overlay Network is functioning once again.

Step 31 Use the following commands to confirm vEdge operation.

```
branch-vE1# show omp peers                    (Output may vary)
R -> routes received
I -> routes installed
S -> routes sent
                          DOMAIN    OVERLAY   SITE
PEER            TYPE      ID        ID        ID        STATE    UPTIME          R/I/S
            ---------------------------------------------------------------------
            -----------------
172.1.0.16      vsmart    1         1         1011      up       0:01:31:57      20/10/4
172.1.0.17      vsmart    1         1         1012      up       0:01:31:55      26/7/4
```

```
branch-vE1# show control connections          (Formatted for
clarity)(Output may vary)
```

```
                                                    PEER                    PEER
PEER    PEER PEER         SITE  DOMAIN  PEER       PRIV PEER        IP      PUB  LOCAL  STATE
TYPE    PROT SYSTEM IP    ID    ID      PRIVATE IP PORT PUBLIC                PORT COLOR
-----------------------------------------------------------------------------------------------
vsmart  dtls 172.1.0.16   1011  1       1.3.26.26  1244 1.3.26.26           12446 gold   up
                                                     6
vsmart  dtls 172.1.0.16   1011  1       1.3.26.26  1244 1.3.26.26           12446 silver up
                                                     6
vsmart  dtls 172.1.0.17   1012  1       1.4.27.27  1244 1.4.27.27           12446 gold   up
                                                     6
vsmart  dtls 172.1.0.17   1012  1       1.4.27.27  1244 1.4.27.27           12446 silver up
                                                     6
vmanage dtls 172.1.0.18   1001  0       1.4.28.28  1284 1.4.28.28           12846 gold   up
                                                     6
```

```
branch-vE1# show ip routes                    (Formatted for
clarity)(Output may vary)
Codes Proto-sub-type:
```

```
              IA -> ospf-intra-area, IE -> ospf-inter-area,
              E1 -> ospf-external1, E2 -> ospf-external2,
              N1 -> ospf-nssa-external1, N2 -> ospf-nssa-external2,
              e -> bgp-external, i -> bgp-internal


            Codes Status flags:
              F -> fib, S -> selected, I -> inactive,
              B -> blackhole, R -> recursive


                                          PROTOCOL NEXTHOP NEXTHOP NEXTHOP
VPN     PREFIX              PROTOCOL      SUB TYPE IF NAME ADDR    VPN TLOC IP
        COLOR               ENCAP STATUS
-------------------------------------------------------------------------------------
 0      0.0.0.0/0             static     - ge0/1   1.1.7.22  -  -           -      -      F,S
 0      0.0.0.0/0             static     - ge0/2   1.2.9.23  -  -           -      -      F,S
 0      1.1.7.0/24            connected  - ge0/1   -         -  -           -      -      F,S
 0      1.2.9.0/24            connected  - ge0/2   -         -  -           -      -      F,S
 0      172.1.10.1/32         connected  - system  -         -  -           -      -      F,S
 1      0.0.0.0/0             omp        - -       -         -  172.1.100.5 gold   ipsec  F,S
 1      0.0.0.0/0             omp        - -       -         -  172.1.100.6 gold   ipsec  F,S
 1      0.0.0.0/0             omp        - -       -         -  172.1.200.7 silver ipsec  F,S
 1      0.0.0.0/0             omp        - -       -         -  172.1.200.8 silver ipsec  F,S
 1      10.2.50.0/24          ospf      IA ge0/0   -         -  -           -      -      -
 1      10.2.50.0/24          connected  - ge0/0   -         -  -           -      -      F,S
 1      10.2.51.0/24          omp        - -       -         -  172.1.20.2  gold   ipsec  F,S
 1      10.2.52.0/24          omp        - -       -         -  172.1.30.3  gold   ipsec  F,S
 1      10.2.53.0/24          omp        - -       -         -  172.1.30.4  gold   ipsec  F,S
 1      10.2.54.0/24          omp        - -       -         -  172.1.100.5 gold   ipsec  F,S
 1      10.2.55.0/24          omp        - -       -         -  172.1.100.6 gold   ipsec  F,S
 1      10.2.56.0/24          omp        - -       -         -  172.1.200.7 silver ipsec  F,S
 1      10.2.57.0/24          omp        - -       -         -  172.1.200.8 silver ipsec  F,S
 1      10.2.58.0/24          omp        - -       -         -  172.1.100.5 gold   ipsec  F,S
 1      10.2.58.0/24          omp        - -       -         -  172.1.100.6 gold   ipsec  F,S
 1      10.2.58.0/24          omp        - -       -         -  172.1.200.7 silver ipsec  F,S
 1      10.2.58.0/24          omp        - -       -         -  172.1.200.8 silver ipsec  F,S
 1      39.39.39.39/32        omp        - -       -         -  172.1.100.5 gold   ipsec  F,S
 1      39.39.39.39/32        omp        - -       -         -  172.1.100.6 gold   ipsec  F,S
 1      99.99.99.99/32        ospf      IA loopback1-        -  -           -      -      -
 1      99.99.99.99/32        connected - loopback1-         -  -           -      -      F,S

            --- VPN 512 OMITTED ---
```

 © 2020 NterOne Corporation

```
branch-vE1# show ipsec local-sa                (Formatted for clarity)(Output may
                                                                  vary)
                                   SOURCE          SOURCE            SOURCE
TLOC ADDRESS  TLOC COLOR   SPI     IPv4            IPv6              PORT    KEY HASH
-----------------------------------------------------------------------------------
-
172.1.10.1    gold         267     1.1.7.11        ::                12346   *****0eb5
172.1.10.1    gold         268     1.1.7.11        ::                12346   *****0eb5
172.1.10.1    silver       267     1.2.9.11        ::                12346   *****2d3f
172.1.10.1    silver       268     1.2.9.11        ::                12346   *****2d3f
```

Activity Verification

You have completed this task when you attain these results:

- You have successfully created a CLI Template using vManage
- You have successfully attached a CLI Template to a vEdge router using vManage

Lab 8: Provision and Deploy Policies (Part 1)

In this Lab, you will provision and deploy Policies on vManage.

- Task 1: Custom Control Policy
- Task 2: Hub and Spoke Control Policy

Activity Objectives

In this activity, you will configure policies to control route distribution and or modification. This policy will help you to learn how to modify control advertisements and route selection.

The following devices and values will be used in this Lab.

Device Name	IP Address or URL	Username	Password
vManage-1	https://1.4.28.28 – SSH 10.82.83.131	admin	admin
Branch-vE1	1.1.7.11 – SSH 10.82.83.111	admin	admin
vSmart-1	1.3.26.26 – SSH 10.82.83.134	admin	admin

VPN List Name	List Value
VPN1	1

Site-list Name	Site-list Values
BRANCHES	10,20,30
DC-HUBS	100,200

 © 2020 NterOne Corporation

Task 1: Modify Controller Distribution using a Control Policy

You will create a Custom Control Policy that will limit route advertisements from vSmart to include HUB Sites only.

Activity Procedure

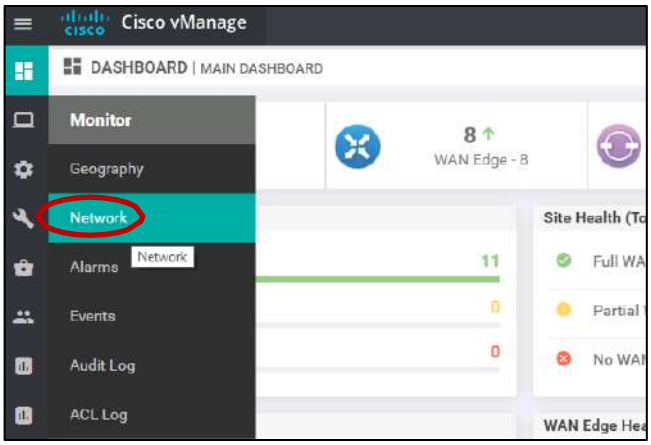
Complete these steps:

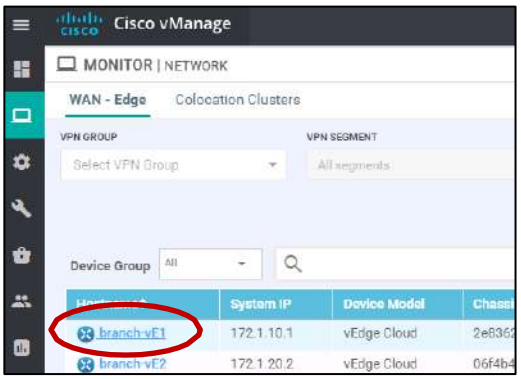
Step 1 Open a browser window and log in to vManage. You will use vManage to create Policies. **A recommendation is to set the resolution on the Admin PC to 1280 X 768 and maximize your browser window.**

Step 2 You will create a **Custom Control Policy** that will limit advertisements from vSmart **(outbound)** to advertise **TLOCs** only from **HUB Sites and no BRANCH Sites** (Hub and Spoke).

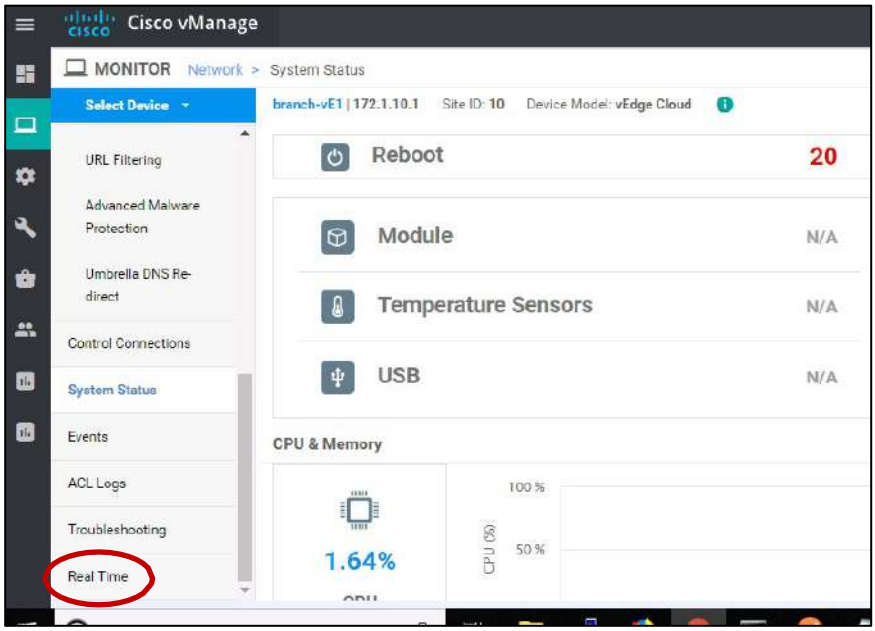
Step 3 First, from vManage verify active **BFD Sessions** on branch-vE1 and verify the **routing table.**

Step 4 Go to **Monitor** and select **Network**, and then click **branch-vE1** from the top left corner of the window.

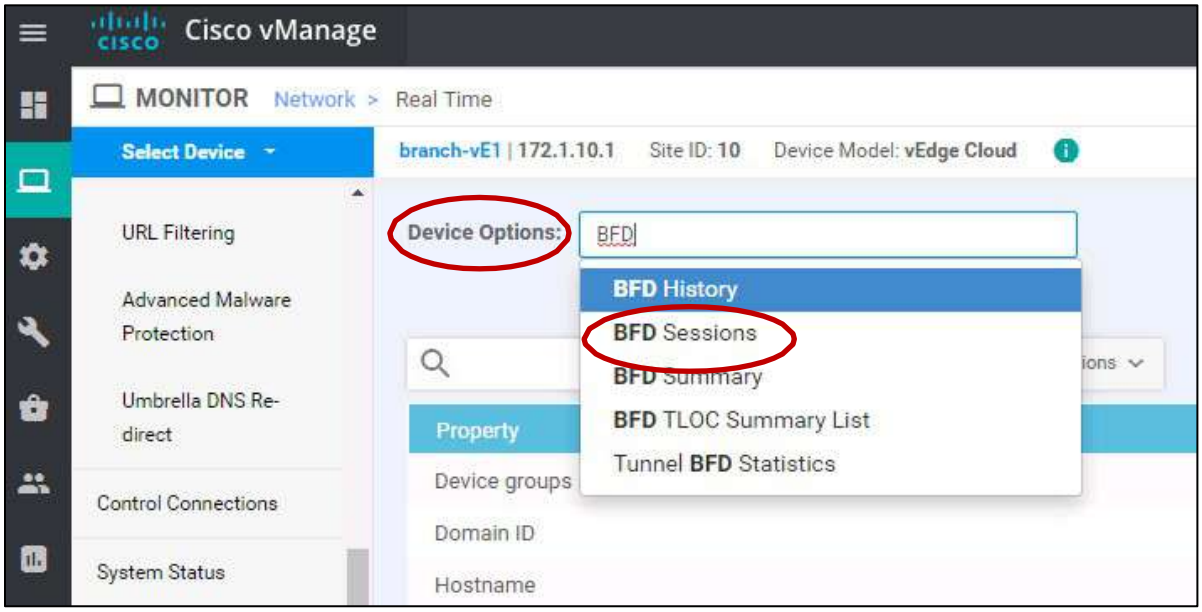




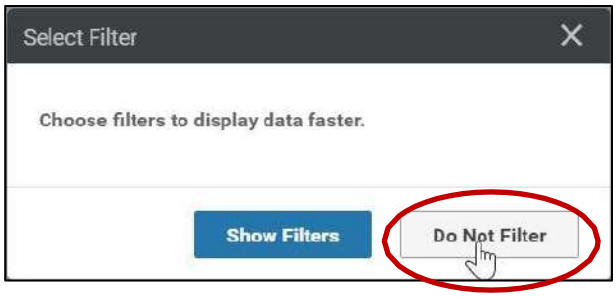
Step 5 Click **Real Time** from the **System Status** category.



Step 6 In the Device Options drop-down select BFD Sessions.



Step 7 Click Do Not Filter.



 © 2020 NterOne Corporation

Step 8 Take note of all the **BFD Sessions to other Branch Routers** (Branch IPsec Tunnels).

branch-vE1 | 172.1.10.1 Site ID: 10 Device Model: vedge-cloud

Device Options: BFD Sessions

Filter

Search Options

System IP	Site ID	State	Source TLOC Color	Remote TLOC Color	Source IP
172.1.20.2	20	up	gold	gold	1.1.7.11
172.1.30.3	30	up	gold	gold	1.1.7.11
172.1.30.4	30	up	gold	gold	1.1.7.11
172.1.100.5	100	up	gold	gold	1.1.7.11
172.1.100.6	100	up	gold	gold	1.1.7.11
172.1.200.7	200	up	gold	silver	1.1.7.11
172.1.200.8	200	up	gold	silver	1.1.7.11
172.1.20.2	20	up	silver	gold	1.2.9.11
172.1.30.3	30	up	silver	gold	1.2.9.11
172.1.30.4	30	up	silver	gold	1.2.9.11
172.1.100.5	100	up	silver	gold	1.2.9.11
172.1.100.6	100	up	silver	gold	1.2.9.11
172.1.200.7	200	up	silver	silver	1.2.9.11
172.1.200.8	200	up	silver	silver	1.2.9.11

Step 9 **Login to your vEdge router** and use the command "**show omp routes | tab**" to verify OMP routes.

Step 10 Advertised **OMP routes** that belong to Branch routers on paths **gold** and **silver** have been **Chosen, Installed, and Resolved (C,I,R).** Currently there are no restrictions on **TLOC advertisements**.

Notice routes from redundant vSmart have been **Chosen, Resolved (C,R),** but not **'I' Installed.**

```
branch-vE1# show omp routes | tab
 Code:
 C   -> chosen
 I   -> installed
 Red -> redistributed
 Rej -> rejected
 L   -> looped
 R   -> resolved
 S   -> stale
 Ext -> extranet
 Inv -> invalid
 Stg -> staged
 U   -> TLOC unresolved
```

```
                                  PATH                ATTRIBUTE
VPN    PREFIX         FROM PEER   ID   LABEL  STATUS  TYPE      TLOC IP      COLOR  ENCAP
 ---------------------------------------------------------------------------------------------
--
 1      0.0.0.0/0      172.1.0.16  21   1002   Inv,U   installed 172.1.100.6  mpls   ipsec
                       172.1.0.16  22   1002   C,I,R   installed 172.1.100.6  gold   ipsec
                       172.1.0.19  24   1002   Inv,U   installed 172.1.100.5  mpls   ipsec
                       172.1.0.19  25   1002   C,I,R   installed 172.1.100.5  gold   ipsec
 1      10.2.50.0/24   0.0.0.0     75   1004   C,Red,R installed 172.1.10.1   gold   ipsec
                       0.0.0.0     76   1004   C,Red,R installed 172.1.10.1   silver ipsec
 1      10.2.51.0/24   172.1.0.19  36   1002   C,I,R   installed 172.1.20.2   gold   ipsec
 1      10.2.54.0/24   172.1.0.19  20   1002   Inv,U   installed 172.1.100.5  mpls   ipsec
                       172.1.0.19  21   1002   C,I,R   installed 172.1.100.5  gold   ipsec
 1      10.2.55.0/24   172.1.0.16  19   1002   Inv,U   installed 172.1.100.6  mpls   ipsec
                       172.1.0.16  20   1002   C,I,R   installed 172.1.100.6  gold   ipsec
 1      10.2.58.0/24   172.1.0.16  25   1002   Inv,U   installed 172.1.100.6  mpls   ipsec
                       172.1.0.16  26   1002   C,I,R   installed 172.1.100.6  gold   ipsec
                       172.1.0.19  28   1002   Inv,U   installed 172.1.100.5  mpls   ipsec
                       172.1.0.19  29   1002   C,I,R   installed 172.1.100.5  gold   ipsec
 1      39.39.39.39/32 172.1.0.16  29   1002   Inv,U   installed 172.1.100.6  mpls   ipsec
                       172.1.0.16  30   1002   C,I,R   installed 172.1.100.6  gold   ipsec
                       172.1.0.19  32   1002   Inv,U   installed 172.1.100.5  mpls   ipsec
                       172.1.0.19  33   1002   C,I,R   installed 172.1.100.5  gold   ipsec
 1      99.99.99.99/32 0.0.0.0     75   1004   C,Red,R installed 172.1.10.1   gold   ipsec
                       0.0.0.0     76   1004   C,Red,R installed 172.1.10.1   silver ipsec
```

Step 11 **Issue** the command **"show ip routes"** to verify the routing table.

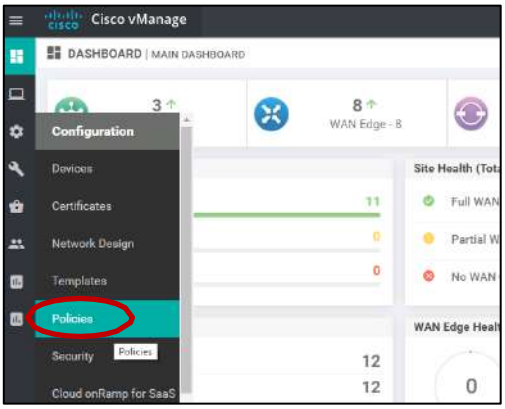
Step 12 Notice that **all routes** from both **Branch routers** and Hub Routers are being advertised and are present in the routing table. This will change once the Control Policy is created and activated.

```
branch-vE1# show ip routes              (Formatted for clarity)(Output
may vary)
Codes Proto-sub-type:
```

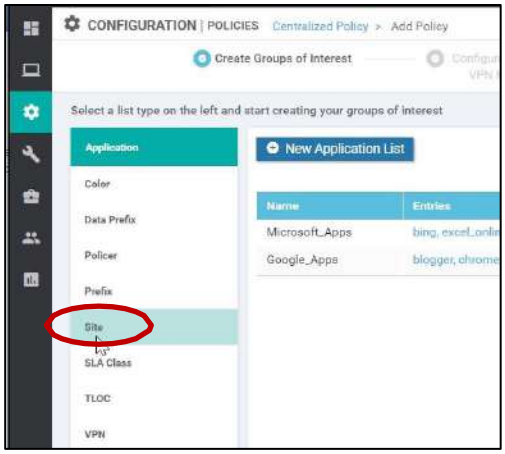
 © 2020 NterOne Corporation

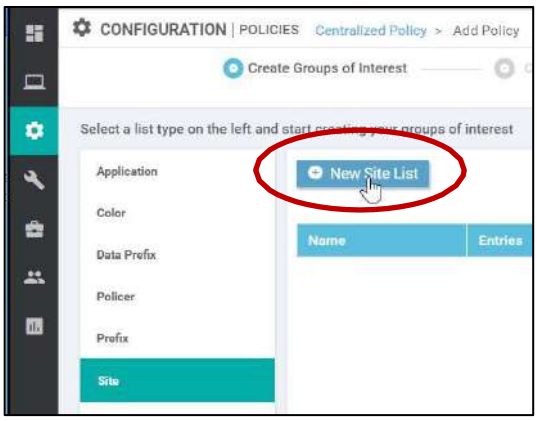
```
                IA -> ospf-intra-area, IE -> ospf-inter-area,
                E1 -> ospf-external1, E2 -> ospf-external2,
                N1 -> ospf-nssa-external1, N2 -> ospf-nssa-external2,
                e -> bgp-external, I -> bgp-internal
              Codes Status flags:
                F -> fib, S -> selected, I -> inactive,
                B -> blackhole, R -> recursive

                                                PROTOCOL NEXTHOP NEXTHOP NEXTHOP
VPN       PREFIX            PROTOCOL            SUB TYPE IF NAME ADDR              VPN TLOC IP
          COLOR ENCAP       STAT
 0        0.0.0.0/0         static     -  ge0/1   1.1.7.22  -  -          -    -      F,S
 0        0.0.0.0/0         static     -  ge0/2   1.2.9.23  -  -          -    -      F,S
 0        1.1.7.0/24        connected  -  ge0/1   -         -  -          -    -      F,S
 0        1.2.9.0/24        connected  -  ge0/2   -         -  -          -    -      F,S
 0        172.1.10.1/32     connected  -  system  -         -  -          -    -      F,S
 1        0.0.0.0/0         omp        -  -       -         -  172.1.100.5 gold ipsec  F,S
 1        0.0.0.0/0         omp        -  -       -         -  172.1.100.6 gold ipsec  F,S
 1        0.0.0.0/0         omp        -  -       -         -  172.1.200.7 silver  ips F,S
                                                                                    e
 1        0.0.0.0/0         omp        -  -       -         -  172.1.200.8 silver  ips F,S
                                                                                    e
 1        10.2.50.0/24      ospf       IA ge0/0   -         -  -          -    -      -
 1        10.2.50.0/24      connected  -  ge0/0   -         -  -          -    -      F,S
 1        10.2.51.0/24      omp        -  -       -         -  172.1.20.2  gold ipsec  F,S
 1        10.2.52.0/24      omp        -  -       -         -  172.1.30.3  gold ipsec  F,S
 1        10.2.53.0/24      omp        -  -       -         -  172.1.30.4  gold ipsec  F,S

--- OUTPUT OMITTED ---
```

Step 13 Once again **Login to vManage** (if not already logged in) and go to **Configuration > Policies,** and then click **Add Policy.** This will start the Wizard to create a **Centralized Policy.**

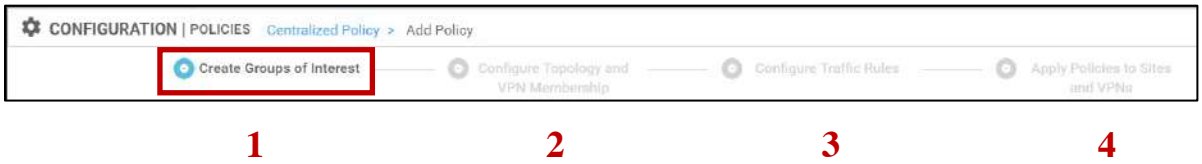


Step 14 Select **Site** from the **List Type** field on the left side of the screen and then click **New Site List.**



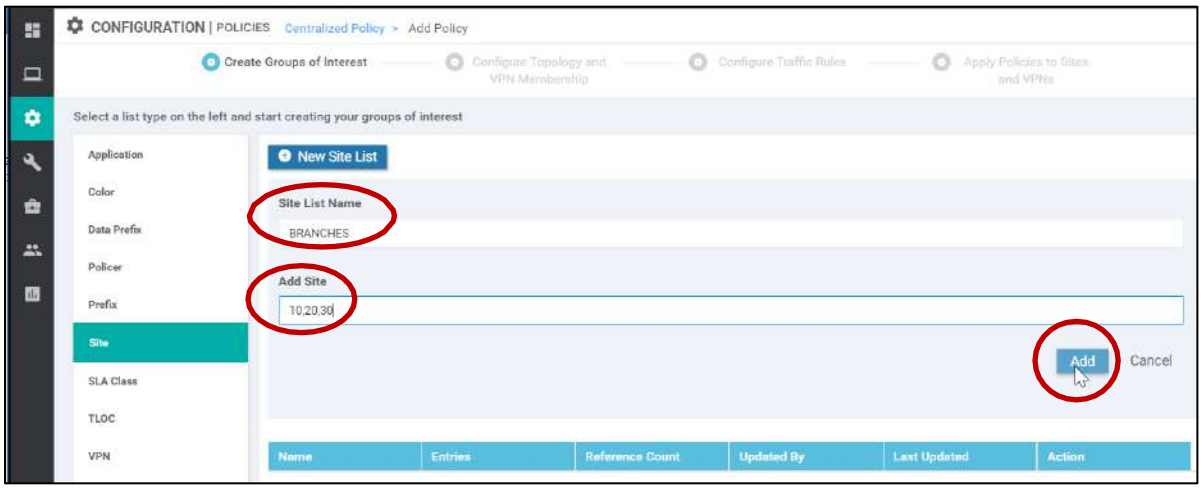


Step 15 The **Centralized Policy** window begins with the Policy **Wizard.** The first step of the wizard allows you to create appropriate **Lists** to be used during policy creation. Whatever lists are created during this step (successfully), will be available regardless whether you cancel and restart the Wizard.



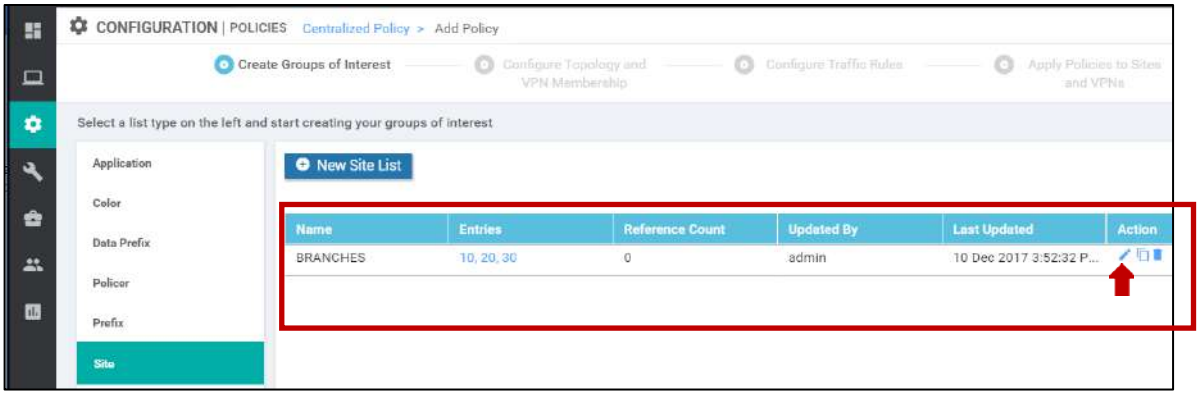
Step 16 Enter the following values and then click **Add**.

- Site List Name: **BRANCHES**
- Add Site: **10,20,30**

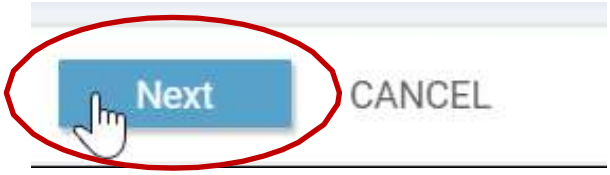


Step 17 Once you've clicked on **Add,** you should see the **Site List Name** window once again. Here you will see the **Site List** you just created along with any other lists that were created previously. If necessary, modify any parameters by using the **pencil** icon located to the **right of the Site List**.

 © 2020 NterOne Corporation

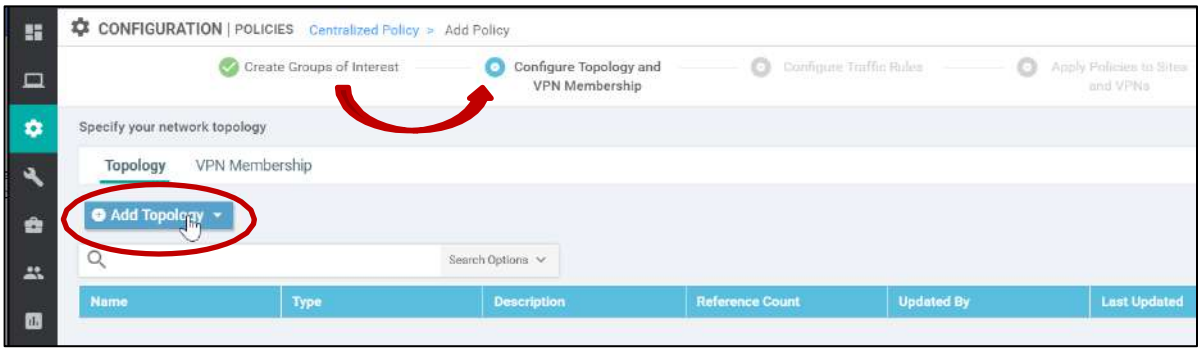


Step 18 Click **Next** located at the bottom of the window.

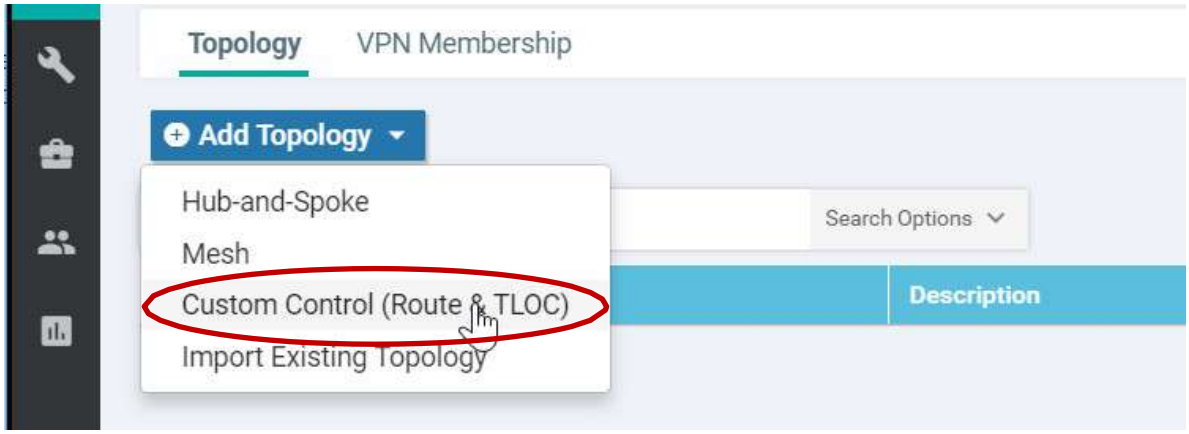


Step 19 At the top of the window notice that the **Wizard Progresses** to the next section.

Click **Add Topology**.

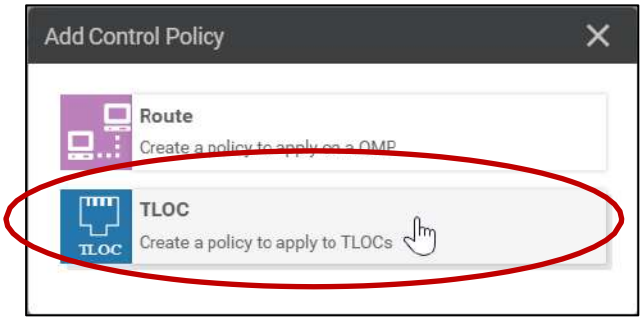


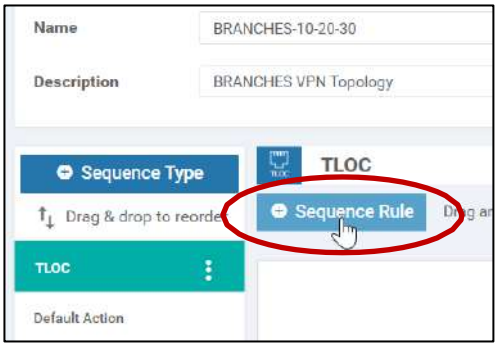
Step 20 Select **Custom Control (Route & TLOC)** from the drop-down menu.



Step 21 Use "**BRANCHES-10-20-30**" for the **Name** and enter a description then click Sequence Type. (**The name here represents the "Network Topology" and the control-policy name in the CLI**)

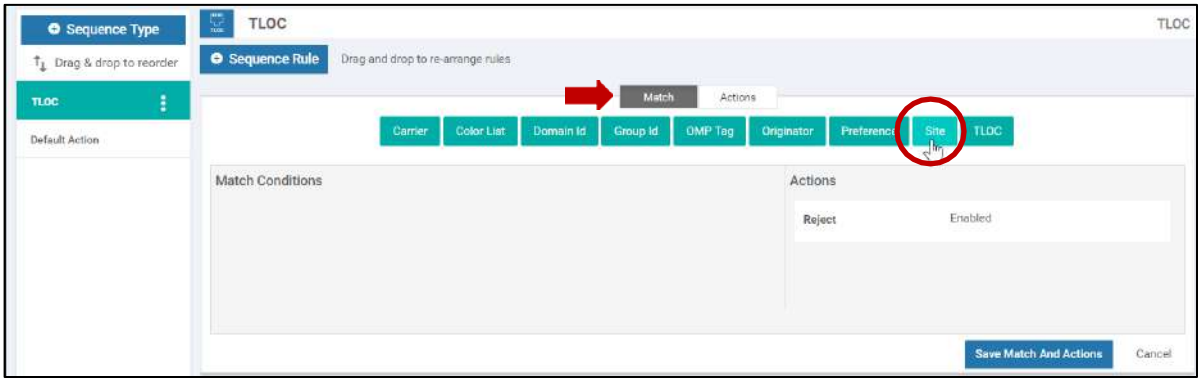
Step 22 Click **TLOC** in the Add Control Policy window and then click + **Sequence Rule**.



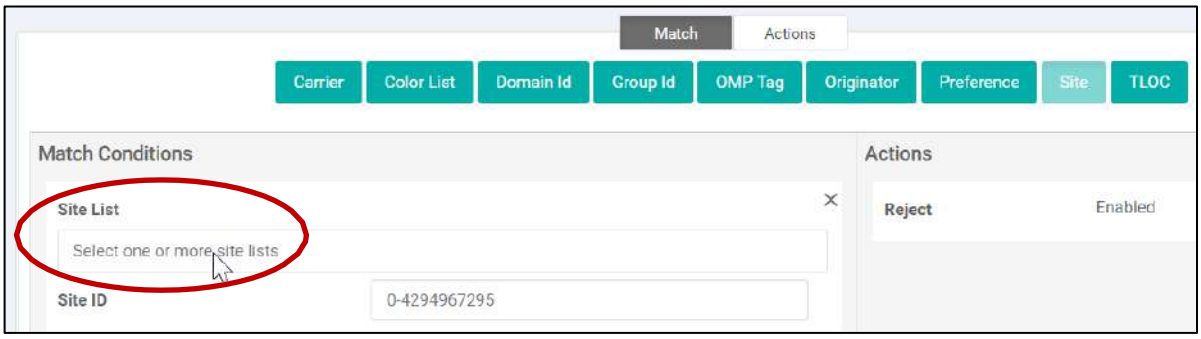


Step 23 The sequence rule should start out with a **Match** (red arrow). Click **Site** from the Match Conditions menu.

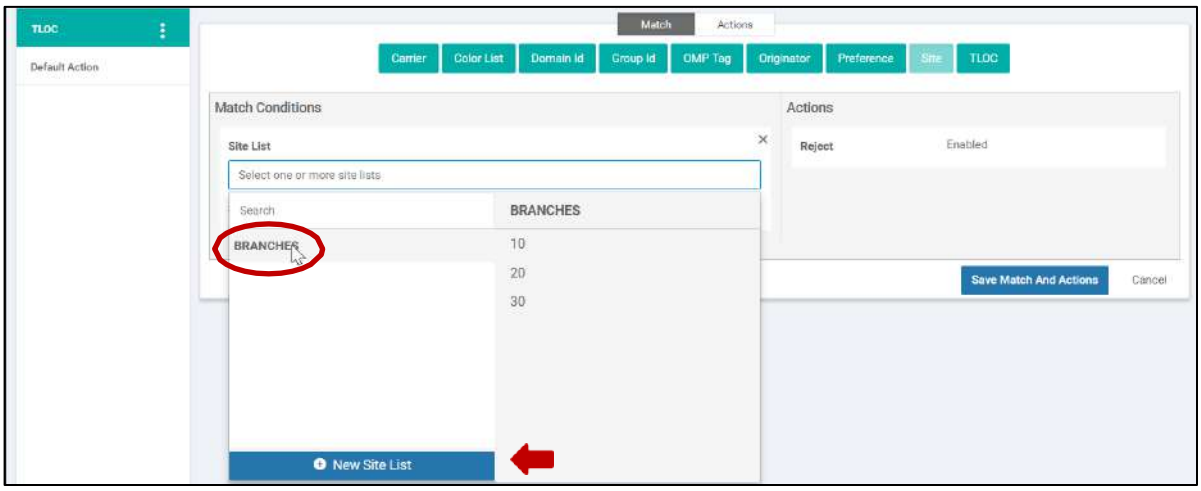
 © 2020 NterOne Corporation



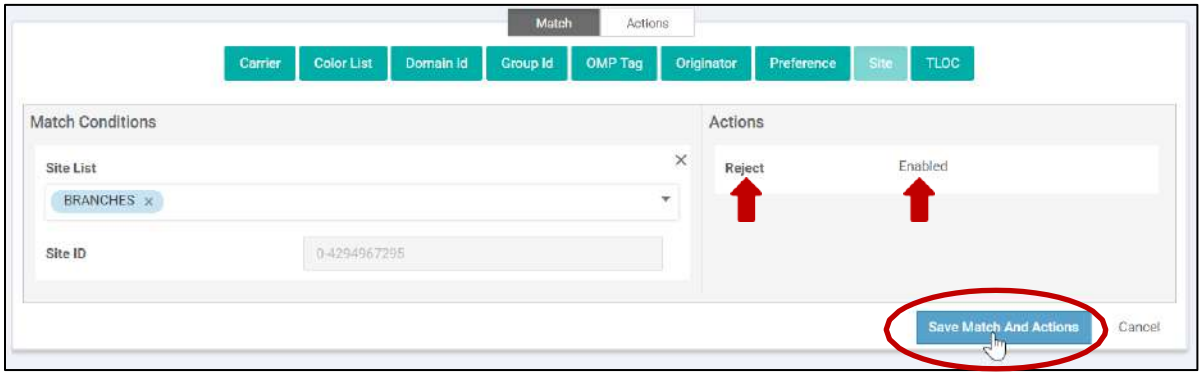
Step 24 Click on Site List from the Match Conditions window.



Step 25 Select **BRANCHES** from the Site List. Notice – you have the opportunity to **create a Site List** if one were not created earlier (red arrow).

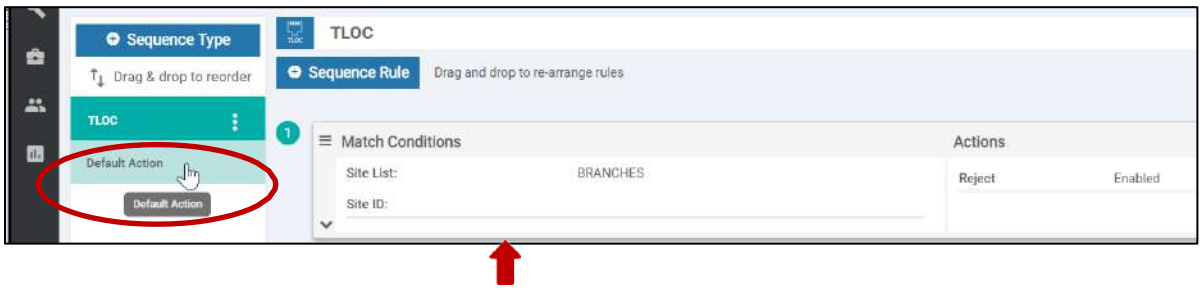


Step 26 The Actions window to the right should read **Reject Enabled** (Default). At the bottom right, click **Save Match And Actions.**



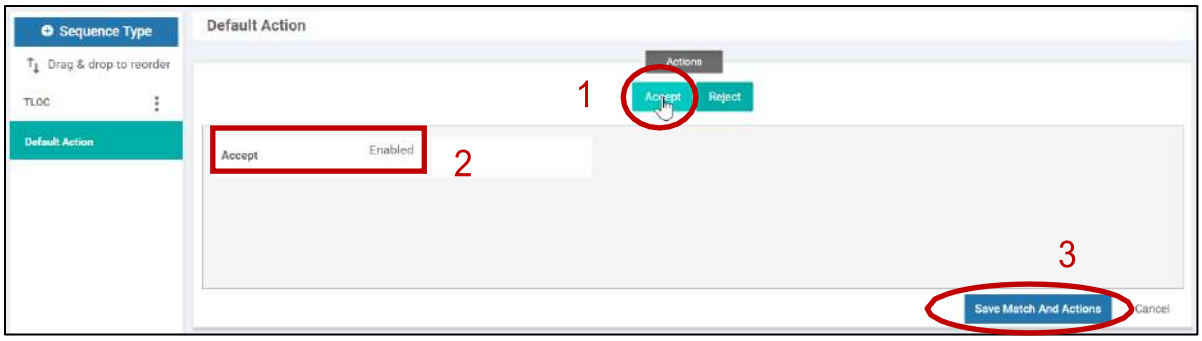
Step 27 You should now have one **Match** Condition on the list (arrow).

Click **Default Action** over to the left side of the window.



Step 28 You have to modify the **Default Action** condition so that it accepts all other advertisements. Click the **pencil** icon over to the right side of the window.

Step 29 Click **Accept**. The Accept Action should now be enabled. At the bottom right, click **Save Match And Actions** to proceed.

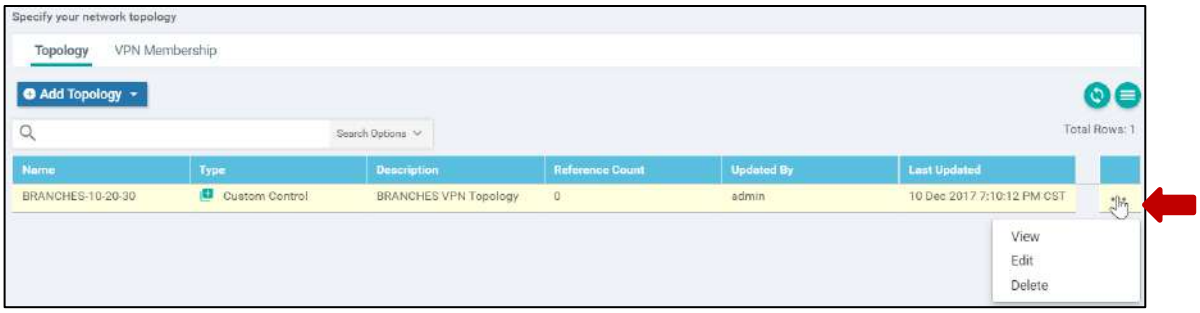


Step 30 The previous Match Sequence will **match any TLOCs** that belong to the Site List **Branches** (Sites – 10,20,30)

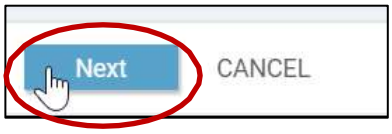
Step 31 Click **Save Control Policy** near the bottom of the window.

 © 2020 NterOne Corporation

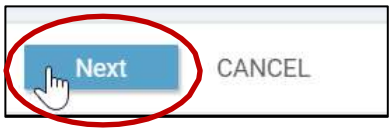
Step 32 Clicking Save will take you back to the **Topology** window. If you need to modify the Topology that was just created **Click …** over to the right side of the window (arrow).



Step 33 Click **Next** located at the bottom center of the window. (We will not be configuring Application Aware Routing at this time. This will be a subsequent lab.)



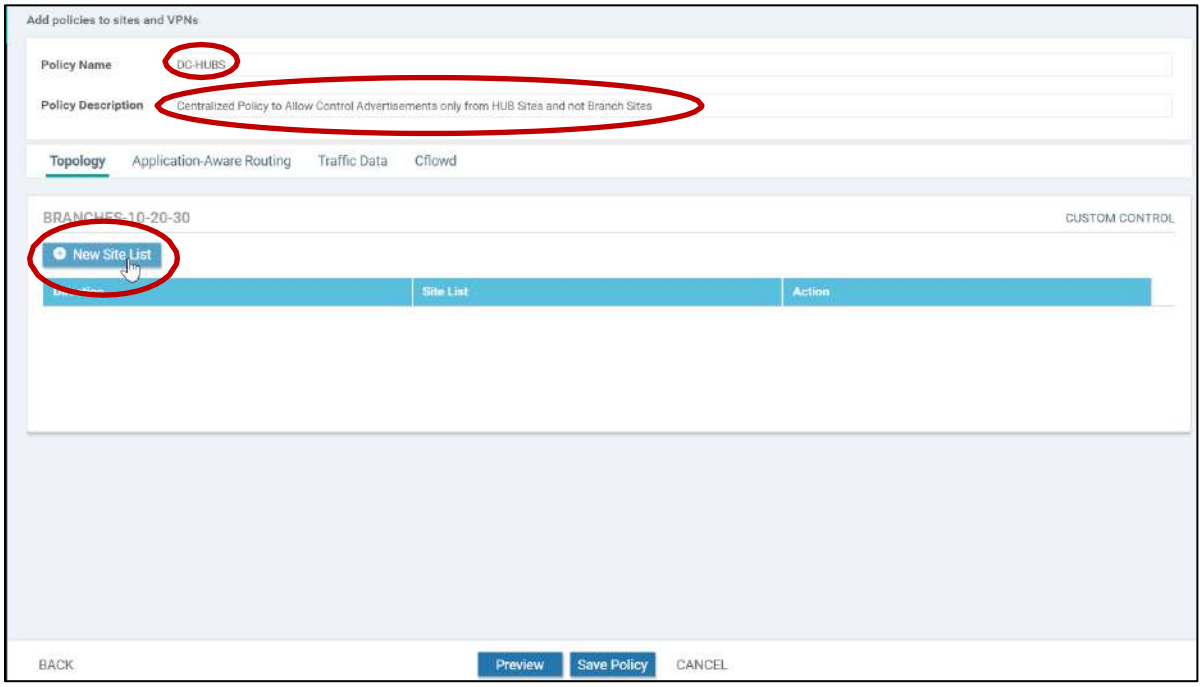
Step 34 Click **Next** once more. You will not create Traffic Rules.



Step 35 This will bring you to the **Apply Policies to Sites and VPNs** window.

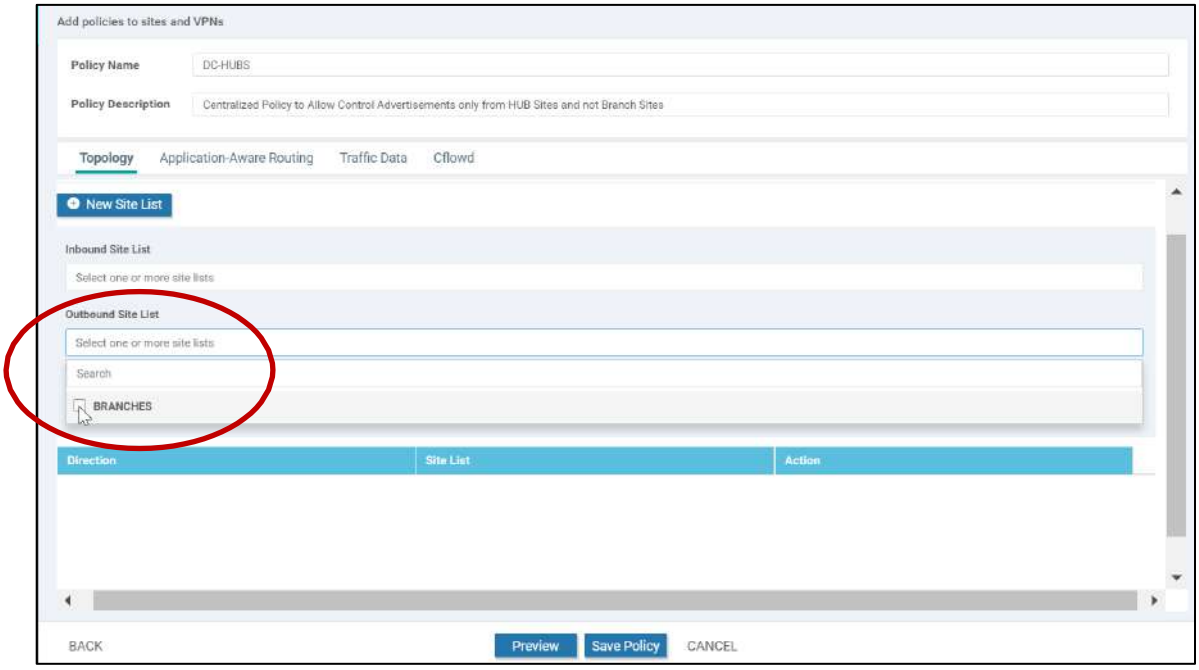
Step 36 Enter **"DC-HUBS"** for **Policy Name** and enter a description. Click **New Site List**. This will allow us to apply the policy to one or more Sites in either the inbound/outbound direction or both.

(This represents the Policy name displayed in the "vManage – Centralized Policy" window)



Step 37 Select **BRANCHES** from the **Outbound Site List** drop-down menu.

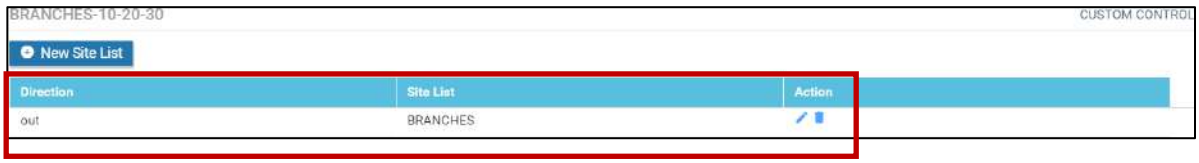
 © 2020 NterOne Corporation



Step 38 Make sure you select the Site List **BRANCHES** for the Outbound Site List. **DO NOT select an Inbound Site List**.

In the bottom right corner, click **Add**.

Step 39 You have now selected a Site List to apply the **DC-HUBS Policy** in the **Outbound direction** (From the vSmart Controllers to the Branches)



Step 40 **(Optional)** If any modifications need to be made to the policy, you can go **BACK** to previous screens, and then use the **pencil** icon to edit any entries before you save the policy.

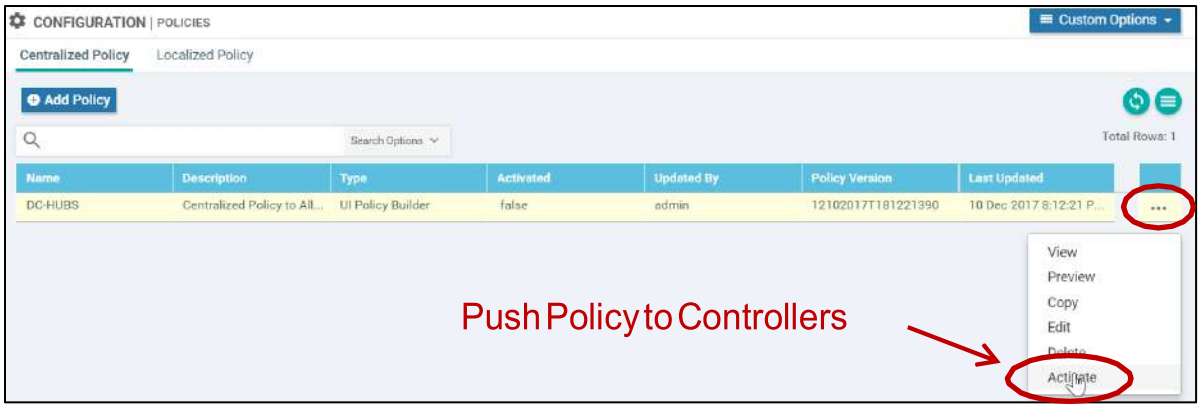
(OPTIONAL)

Step 41 Click **Preview** and verify with the output shown below. Go back if you need to edit any entries.

```
  policy
control-policy BRANCHES-10-20-30
   sequence 1
    match tloc
     site-list BRANCHES
    !
    action reject
    !
   !
  default-action accept
!
lists
 site-list BRANCHES
   site-id 10
   site-id 20
   site-id 30
  !
 !
!
apply-policy
 site-list BRANCHES
  control-policy BRANCHES-10-20-30 out
 !
!
```

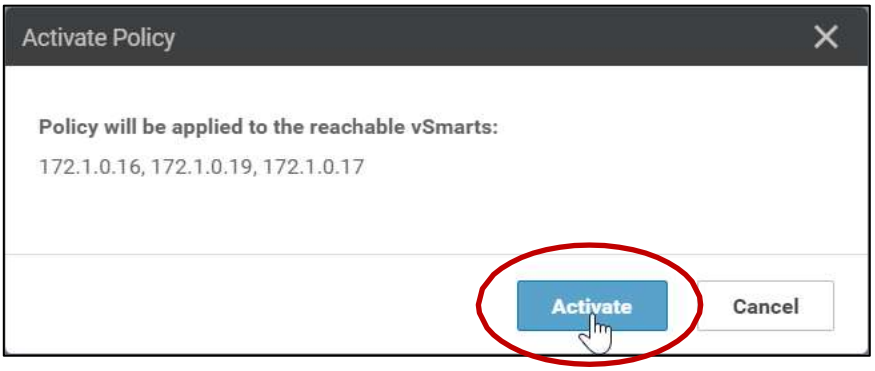
Step 42 Click Save Policy.

Step 43 You'll be redirected to the Policies Main window. Click the **…** to the right of the window, and then click **Activate** to activate this policy and **push it to the vSmart Controllers**. **A modified policy will likely need to be deactivated, before it can be activated.**



 © 2020 NterOne Corporation

Step 44 Click **Activate** to apply the Policy to the listed **vSmart Controllers.**

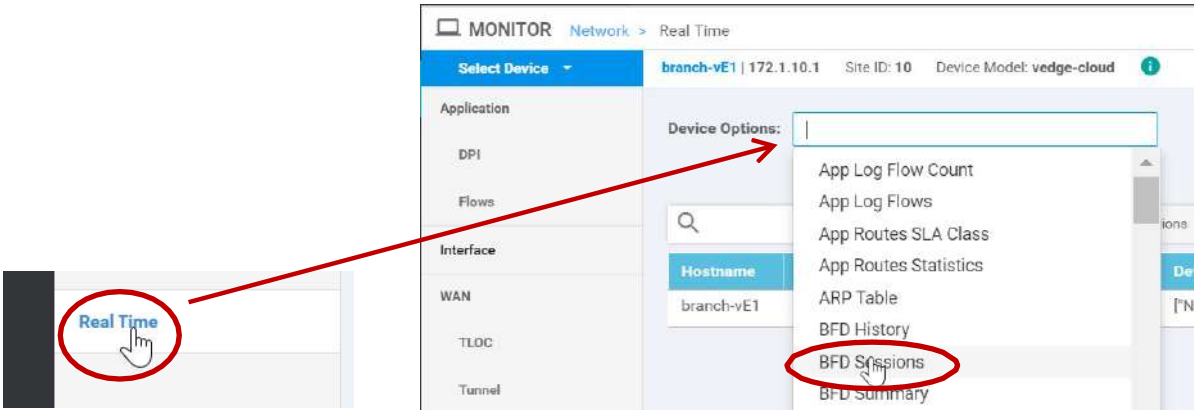


Step 45 You will be redirected to the **Tasks View** window to verify task success. **Contact your Instructor if any of the tasks fail**.

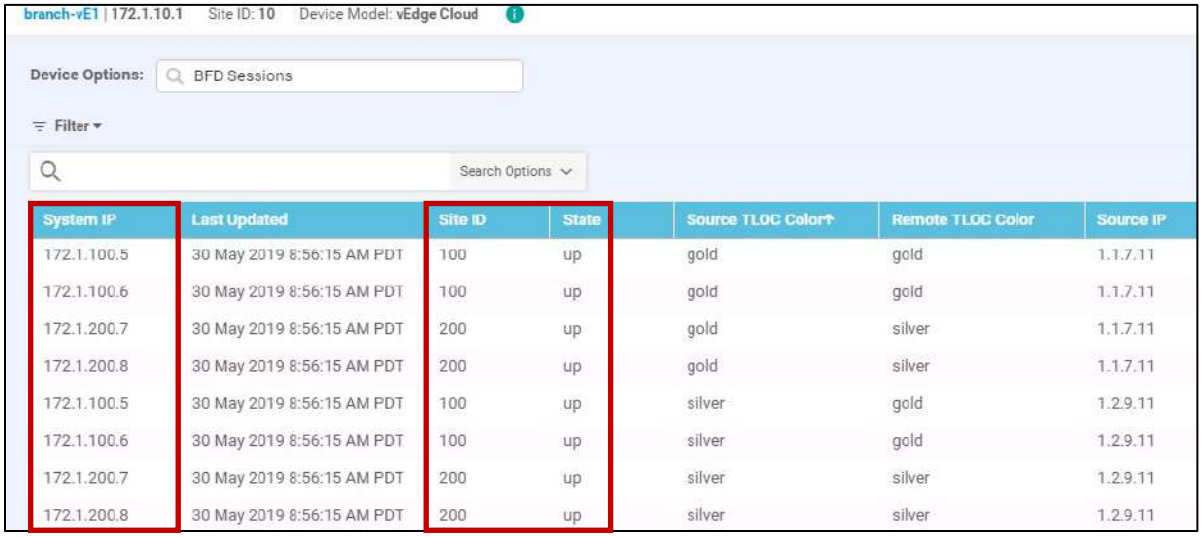
Step 46 Once the Policy has been activated go back and verify the **Branch routers BFD Sessions**.

Step 47 Go to **Monitor > Network** and select **branch-vE1** from the list of devices.

Step 48 Select **Real Time** and then select **BFD Sessions** from the **Device Options** drop-down menu.



Step 49 Click **Do Not Filter.** Verify that the only BFD sessions from your Branch router are the ones to the HUB routers (172.1.100.5, 172.1.100.6 and 172.1.200.7, 172.1.200.8)



System IP	Last Updated	Site ID	State	Source TLOC Color	Remote TLOC Color	Source IP
172.1.100.5	30 May 2019 8:56:15 AM PDT	100	up	gold	gold	1.1.7.11
172.1.100.6	30 May 2019 8:56:15 AM PDT	100	up	gold	gold	1.1.7.11
172.1.200.7	30 May 2019 8:56:15 AM PDT	200	up	gold	silver	1.1.7.11
172.1.200.8	30 May 2019 8:56:15 AM PDT	200	up	gold	silver	1.1.7.11
172.1.100.5	30 May 2019 8:56:15 AM PDT	100	up	silver	gold	1.2.9.11
172.1.100.6	30 May 2019 8:56:15 AM PDT	100	up	silver	gold	1.2.9.11
172.1.200.7	30 May 2019 8:56:15 AM PDT	200	up	silver	silver	1.2.9.11
172.1.200.8	30 May 2019 8:56:15 AM PDT	200	up	silver	silver	1.2.9.11

Step 50 **Login to your branch-vE1 router** and use the command **"show ip route"** to verify the routing table. Notice that the only routes received are being **advertised by HUB routers**. The Branch routes have been removed as a result of the **Control Policy that was activated.** Compare this output to the output on page 131 in the lab guide.

```
branch-vE1# show ip route                    (Formatted for clarity)
Codes Proto-sub-type:
  IA -> ospf-intra-area, IE -> ospf-inter-area,
  E1 -> ospf-external1, E2 -> ospf-external2,
  N1 -> ospf-nssa-external1, N2 -> ospf-nssa-external2,
  e -> bgp-external, I -> bgp-internal
Codes Status flags:
  F -> fib, S -> selected, I -> inactive,
  B -> blackhole, R -> recursive
```

 © 2020 NterOne Corporation

```
                                        PROTOCOL NEXTHOP NEXTHOP NEXTHOP
VPN     PREFIX          PROTOCOL  SUB TYPE IF NAME ADDR     VPN TLOC IP     COLOR ENCAP
STATUS
0       0.0.0.0/0           static    -  ge0/1   1.1.7.22  -  -            -     -       F,S
0       0.0.0.0/0           static    -  ge0/2   1.2.9.23  -  -            -     -       F,S
0       1.1.7.0/24          connected -  ge0/1   -         -  -            -     -       F,S
0       1.2.9.0/24          connected -  ge0/2   -         -  -            -     -       F,S
0       172.1.10.1/32       connected -  system  -         -  -            -     -       F,S
        0.0.0.0/0           omp ----------------------------------------172.1.100.5 gold  ipsec  F,S  gold
1       0.0.0.0/0           omp ----------------------------------------172.1.100.6 ipsec F,S  silver ipse
1       0.0.0.0/0           omp ----------------------------------------172.1.200.7 F,S
1       0.0.0.0/0           omp ----------------------------------------172.1.200.8 silver ipse F,S
1       10.2.50.0/24        ospf         IA ge0/0     -         -  -            -     -       -
1       10.2.50.0/24        connected    -  ge0/0     -         -  -            -     -       F,S
1       10.2.54.0/24        omp          -  -         -         -  172.1.100.5  gold  Ipsec   F,S
1       10.2.55.0/24        omp          -  -         -         -  172.1.100.6  gold ipsec   F,S
1       10.2.56.0/24        omp          -  -         -         -  172.1.200.7  silver ipse F,S
1       10.2.57.0/24        omp          -  -         -         -  172.1.200.8  silver ipse F,S
1       39.39.39.39/32      omp          -  -         -         -  172.1.100.5  gold ipsec   F,S
1       39.39.39.39/32      omp          -  -         -         -  172.1.100.6  gold ipsec   F,S
1       99.99.99.99/32      ospf         IA loopback1-          -  -            -     -       -
1       99.99.99.99/32      connected -  loopback1-            -  -            -     -       F,S

--- VPN 512 OMITTED ---
```

Step 51 Issue the command **"show omp routes"** to verify OMP routes.

Step 52 Notice that all of the OMP routes advertised from Branch routers are now marked as **Invalid** and **TLOC Unresolved.** ***(This is the reason that the Branch routes were not installed – Inaccessible TLOC)***

```
branch-vE1# show omp routes                (Formatted for Clarity)(Output may vary)
Code:
C   -> chosen
I   -> installed
Red -> redistributed
Rej -> rejected
L   -> looped
R   -> resolved
S   -> stale
Ext -> extranet
Inv -> invalid
U   -> TLOC unresolved
```

VPN	PREFIX	FROM PEER	ID	LABEL	PATH STATUS	ATTRIBUTE TYPE	TLOC IP	COLOR	ENCAP	PREF
1	0.0.0.0/0	172.1.0.16	54	1002	Inv,U	installed	172.1.100.5	mpls	ipsec	-
		172.1.0.16	55	1002	C,I,R	installed	172.1.100.5	gold	ipsec	-
		172.1.0.16	56	1002	Inv,U	installed	172.1.100.6	mpls	ipsec	-
		172.1.0.16	57	1002	C,I,R	installed	172.1.100.6	gold	ipsec	-
		172.1.0.16	58	1002	Inv,U	installed	172.1.200.8	mpls	ipsec	-
		172.1.0.16	59	1002	C,I,R	installed	172.1.200.8	silver	ipsec	-
		172.1.0.16	68	1002	Inv,U	installed	172.1.200.7	mpls	ipsec	-
		172.1.0.16	69	1002	C,I,R	installed	172.1.200.7	silver	ipsec	-
1	10.2.51.0/24	172.1.0.16	40	1002	Inv,U	installed	172.1.20.2	mpls	ipsec	-
		172.1.0.16	41	1002	Inv,U	installed	172.1.20.2	gold	ipsec	-
1	10.2.52.0/24	172.1.0.16	74	1002	Inv,U	installed	172.1.30.3	mpls	ipsec	-
		172.1.0.16	75	1002	Inv,U	installed	172.1.30.3	gold	ipsec	-
1	10.2.53.0/24	172.1.0.16	60	1002	Inv,U	installed	172.1.30.4	gold	ipsec	-
		172.1.0.16	62	1002	Inv,U	installed	172.1.30.4	mpls	ipsec	-
1	10.2.54.0/24	172.1.0.16	38	1002	Inv,U	installed	172.1.100.5	mpls	ipsec	-
		172.1.0.16	39	1002	C,I,R	installed	172.1.100.5	gold	ipsec	-
1	10.2.55.0/24	172.1.0.16	44	1002	Inv,U	installed	172.1.100.6	mpls	ipsec	-
		172.1.0.16	45	1002	C,I,R	installed	172.1.100.6	gold	ipsec	-
1	10.2.56.0/24	172.1.0.16	66	1002	Inv,U	installed	172.1.200.7	mpls	ipsec	-
		172.1.0.16	67	1002	C,I,R	installed	172.1.200.7	silver	ipsec	-
1	10.2.57.0/24	172.1.0.16	52	1002	Inv,U	installed	172.1.200.8	mpls	ipsec	-
		172.1.0.16	53	1002	C,I,R	installed	172.1.200.8	silver	ipsec	-
1	39.39.39.39/32	172.1.0.16	70	1002	Inv,U	installed	172.1.100.5	mpls	ipsec	-
		172.1.0.16	71	1002	C,I,R	installed	172.1.100.5	gold	ipsec	-
		172.1.0.16	72	1002	Inv,U	installed	172.1.100.6	mpls	ipsec	-
		172.1.0.16	73	1002	C,I,R	installed	172.1.100.6	gold	ipsec	-
1	99.99.99.99/32	0.0.0.0	46	1002	C,Red,R	installed	172.1.10.1	gold	ipsec	-
		0.0.0.0	47	1002	C,Red,R	installed	172.1.10.1	silver	ipsec	-

Step 53 **Login** to the **vSmart-1** Controller and use the command **"show running-config policy"** to verify the Policy.

```
vSmart-1# show running-config policy
   policy
    lists
     site-list BRANCHES
      site-id 10
      site-id 20
      site-id 30
   !
   !
   control-policy BRANCHES-10-20-30
```

 © 2020 NterOne Corporation

```
 sequence 1
   match tloc
    site-list BRANCHES
   !
   action reject
   !
  !
  default-action accept
 !
!
```

Step 54 End of Task 1.

Task 2: Create a Hub and Spoke Control Policy

You will use the built-in **Hub and Spoke Control Policy** that limits **TLOCs & Routes** from vSmart to HUB Sites only. This is a **Challenge Lab.** Use your experience from the previous lab to complete.

Activity Procedure

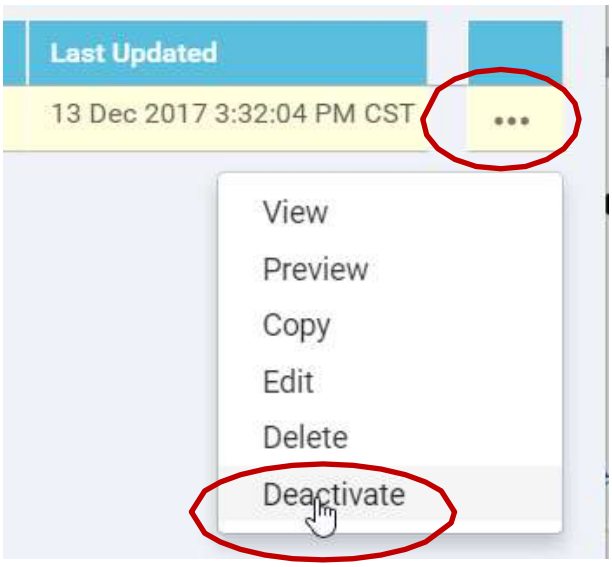
Complete these steps:

Step 1 Open a browser window and log in to vManage. You will use vManage to create Policies.

Step 2 You will create a **Hub and Spoke Control Policy** that will limit advertisements from vSmart **(outbound)** to advertise **TLOCs** and **Routes** only from **HUB Sites.**

Step 3 **Go to** Configuration > Policies **>** Centralized Policy

Step 4 Make sure to **Deactivate** any previous Centralized Policies that are still in effect.



Step 5 Use **show ip routes** to verify the routing table on **branch-vE1.** Confirm that **all routes are present** (from both Hub and Branch sites)

Step 6 In vManage, go back to the **Configuration > Policies**.

Step 7 Click **Add Policy** from the top left corner or mid window if there are no Policies.

Step 8 Create a new **Site List** named **DC-HUBS** and use sites **100,200** (with comma). Do not click **Next**.

Step 9 Create a VPN List **VPN1** with the value **1**. Click **Add**.

Step 10 Click **Next**

Step 11 Click **Add Topology** and select Hub and Spoke

Step 12 Provide the following:

- Name: **HUBS**
- Description: **Hub and Spoke Topology**
- VPN List: **VPN1**

Step 13 **Click Add Hub-and-Spoke.** (Left side)

Step 14 Click **Add Hub Sites** (select DC-HUBS) and then click **Add**.

Step 15 Click **Add Spoke Sites** (select **BRANCHES** from previous Lab) and then click **Add.**

Step 16 **Click** Save Hub-and-Spoke Policy.

Step 17 Click **Next**.

Step 18 Click **Next.** (We will not be configuring Application Aware Routing at this time).

Step 19 Provide a Policy Name and Policy Description.

 © 2020 NterOne Corporation

Step 20 Click **Preview** and verify with output shown below. If necessary go **BACK** and make changes.

```
policy
 control-policy control_359610727
    sequence 10
     match route
      site-list DC-HUBS
      vpn-list VPN1
     !
     action accept
     !
    sequence 20
     match route
      site-list DC-HUBS
      vpn-list VPN1
     !
     action accept

    !
    sequence 30
     match tloc
      site-list DC-HUBS
     !
     action accept
     !
    !
 default-action reject
!
lists
 site-list BRANCHES
  site-id 10
  site-id 20
  site-id 30
 !
 site-list DC-HUBS
  site-id 100
  site-id 200
 !
 vpn-list VPN1
  vpn 1
  !
 !
!
```

```
apply-policy
 site-list BRANCHES
  control-policy control_359610727 out
 !
!
```

Step 21 Click **Save Policy**

Step 22 **Activate** Policy.

Step 23 On **branch-vE1** verify the routing table and confirm that only Hub routes are present.

Step 24 Use the command "**show omp routes**" and "**show omp tlocs**" to verify that there are no TLOCs from Branch routers.

Step 25 End of Task 2.

 © 2020 NterOne Corporation

Lab 9: Provision and Deploy Policies (Part 2)

In this Lab, you will provision and deploy Policies on vManage.

- Task 1: Create an Application Aware Routing Policy
- Task 2: Create a Service Chain Policy
- Task 3: Create Localized QOS Policies Using Feature Templates

Activity Objectives

In this activity, you will configure policies to control tunnel selection, implement a Service chain and create a localized QOS policy.

The following devices and values will be used in this Lab.

Device Name	IP Address or URL	Username	Password
vManage-1	https://1.4.28.28 – SSH 10.82.83.131	admin	admin
branch-vE1	1.1.7.11 – SSH 10.82.83.111	admin	admin
branch-vE2	1.1.10.12 – SSH 10.82.83.112	admin	admin
vSmart-1	1.3.26.26 – SSH 10.82.83.134	admin	admin
vSmart-2	1.4.27.27 – SSH 10.82.83.135	admin	admin

VPN List Name	List Value
VPN1	1

Site-list Name	List Value
SITE-10	10

SLA Class	Latency Value
LAT-50	50ms

Data Prefix List Name	Prefix Value
PREFIX-10-2-58-0_24	10.2.58.0/24
PREFIX-39-39-39-39_32	39.39.39.39/32

Prefix List Name	Prefix Value
PREFIX-10-2-54-0_24	10.2.54.0/24

Service	Service IP	Service TLOC	Service VPN	Color	Encap
Firewall	10.2.51.19	172.1.20.2	1	gold	IPSEC

Task 1: Create an Application Aware Routing Policy

You will create an Application Aware Routing Policy that chooses a tunnel based on the criteria of an SLA class.

Activity Procedure

Complete these steps:

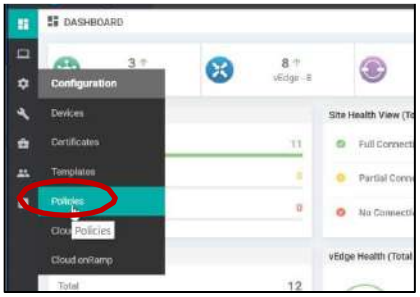
Step 1 Open a browser window and **login to vManage**. You will use vManage to create Policies.

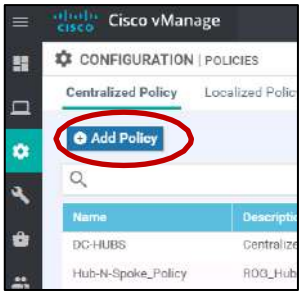
Step 2 You will create an **Application Aware Routing Policy** that will selectively use a Tunnel Interface, depending upon the SLA criteria to which it is compared.

 © 2020 NterOne Corporation

Step 3 Go to **Configuration > Policies,** and then click **Add Policy.** This will start the Wizard to create a Centralized Policy.

1 OR 2

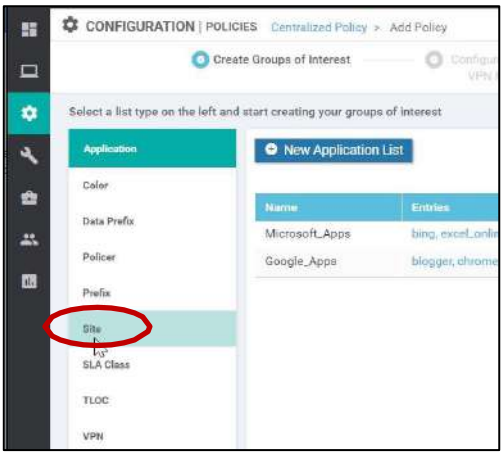


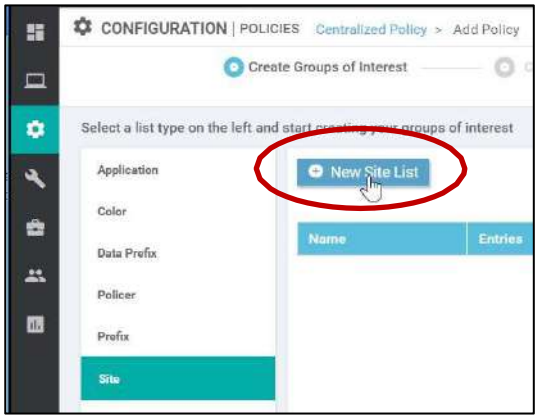


Step 4 This will start the Policy **Wizard.** The first section of the wizard allows you to create **Lists** to be used during policy creation. Whatever lists are created during this step (successfully) the lists will be available regardless whether you cancel and restart the Wizard.

1 2 3 4

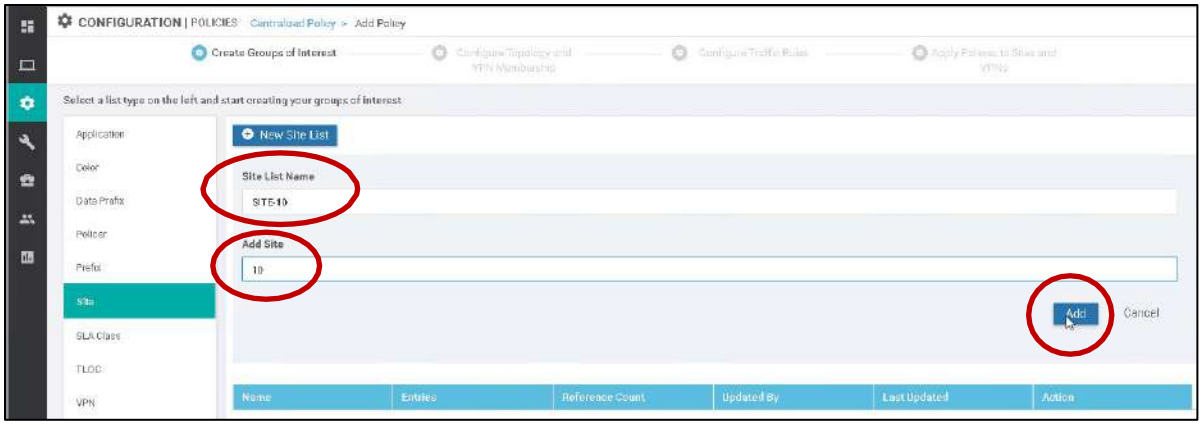
Step 5 Select **Site** from the **List Type** field on the left side of the screen and then click **+ New Site List.**





Step 6 Enter the following values and then click **Add**.

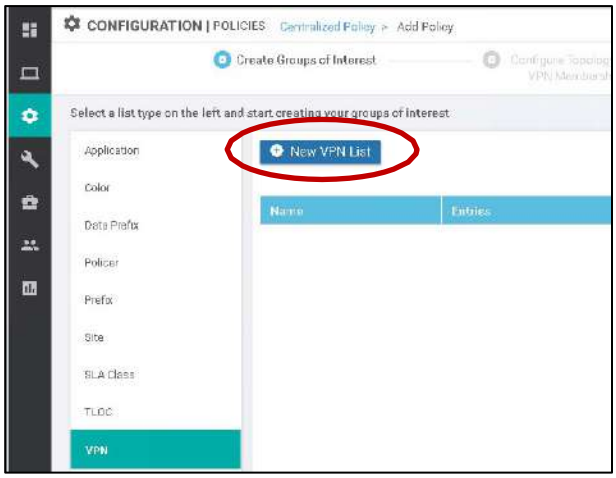
- Site List Name: **SITE-10**
- Add Site: **10**



Step 7 Once you've clicked on **Add** you should see the **Site List** window once again.

Here you will see the List you just created along with any other lists that were created previously. If necessary modify any parameters by using the **pencil** icon located to the **right of the List**.

Step 8 Select **VPN** from the **List Type** field on the left side of the screen and then click **+ New VPN List.**

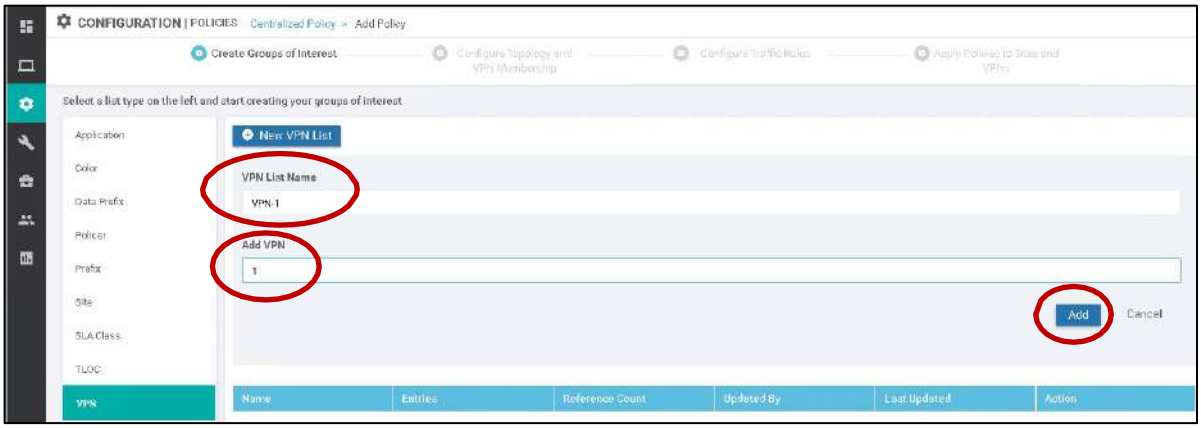


Step 9 Enter the following values and then click **Add** (skip this step if already present).

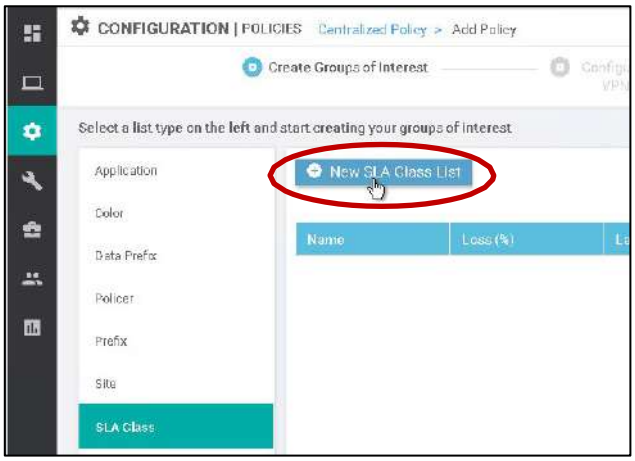
- VPN List Name: **VPN1**

 © 2020 NterOne Corporation

- Add VPN: **1**



Step 10 Select **SLA Class** from the **List Type** field on the left side of the screen and then click **New SLA Class List.**



Step 11 Enter the following values and then click **Add**.

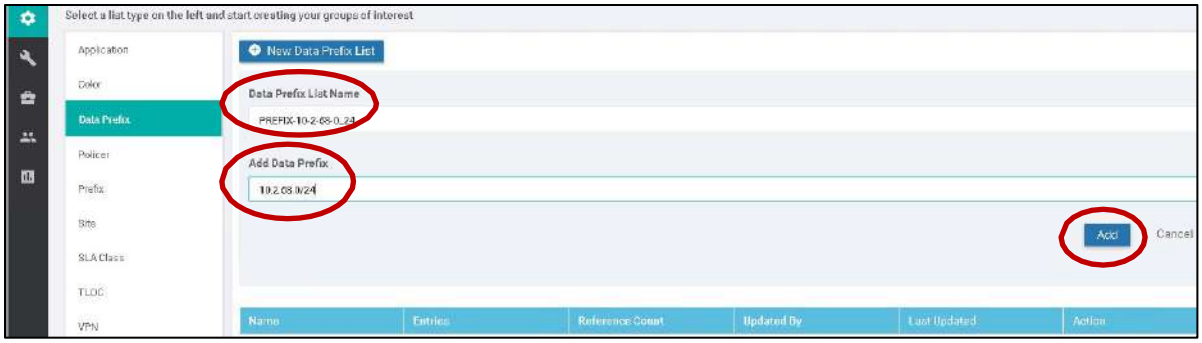
- SLA Class List Name: **LAT-50**
- Latency (ms): **50**

Click **Add.**

Step 12 Select **Data Prefix** on the left side of the screen and then click **+ New Data Prefix List.**

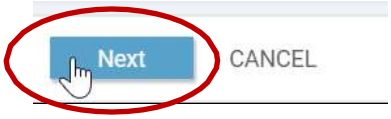
Step 13 Enter the following values and then click **Add**.

- Data Prefix List Name: **PREFIX-10-2-58-0_24**
- Add Data Prefix: **10.2.58.0/24**



Step 14 Create one more Data Prefix List named PREFIX-39-39-39-39_32 with the value 39.39.39.39/32.

Click **Add** when done.

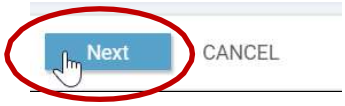


Step 15 Click **Next** to continue.

Step 16 At the top of the window notice that the **Wizard Progresses** to the next section.

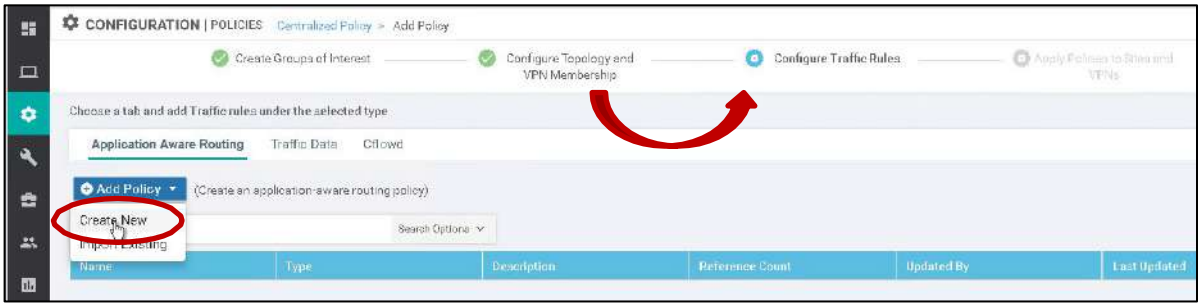
 © 2020 NterOne Corporation

Step 17 Click **Next** to continue.

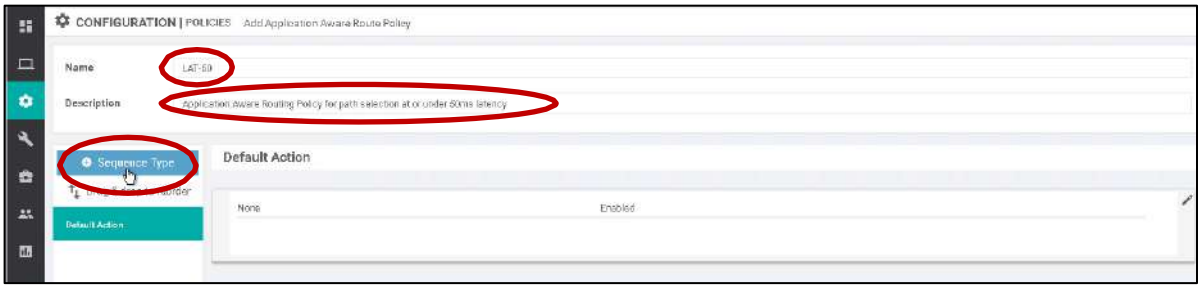


You will not create a Topology or VPN Membership Policy during this task.

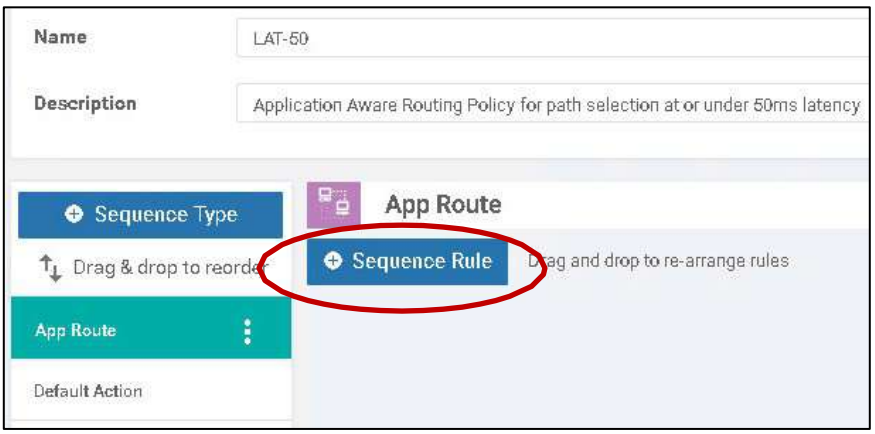
Step 18 The Wizard will progress to the **Configure Traffic Rules** section - Application Aware Routing by default. Click **+ Add Policy** and select **Create New** from the drop-down menu.



Step 19 Use "**LAT-50**" for the Application Aware Routing Policy Name and enter a Description, and then click **+ Sequence Type** on the left side.

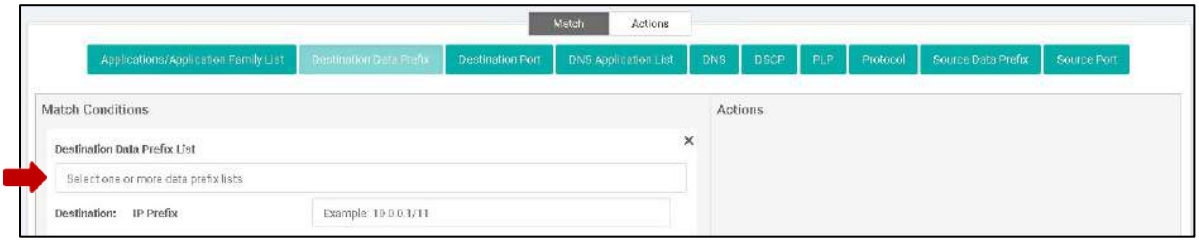


Step 20 Click **+ Sequence Rule**.



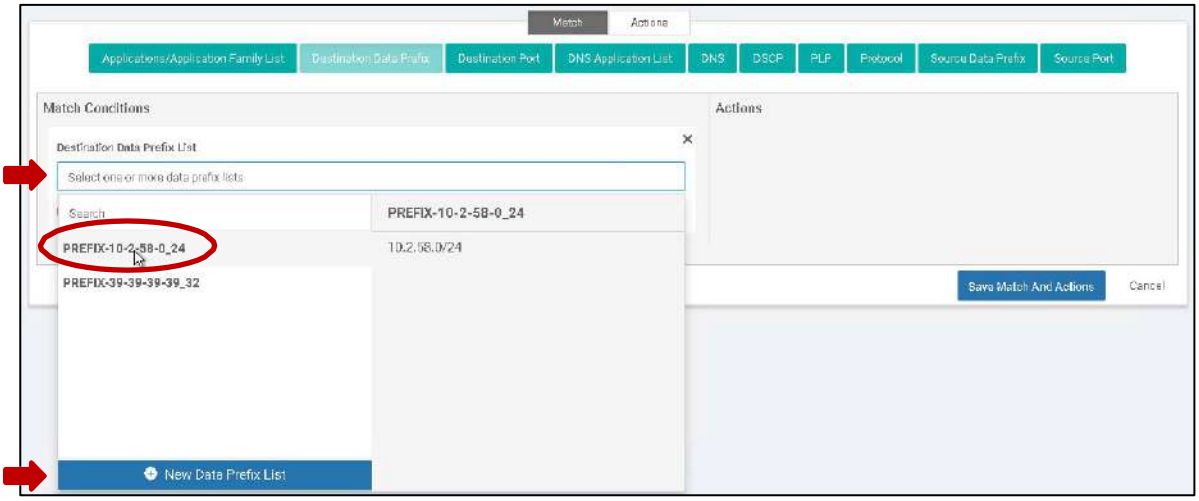
Step 21 The Sequence Rule should start out with a **Match** (arrow). Click **Destination Data Prefix** from the Match Conditions menu.

Step 22 Click on **Destination Data Prefix List** (arrow) from the **Match Conditions** window.



Step 23 Select PREFIX-10-2-58-0_24 from the Destination Data Prefix List drop-down menu (1st arrow).

You have the opportunity to **create a** + **New Data Prefix List** if one were not created earlier (2nd arrow).



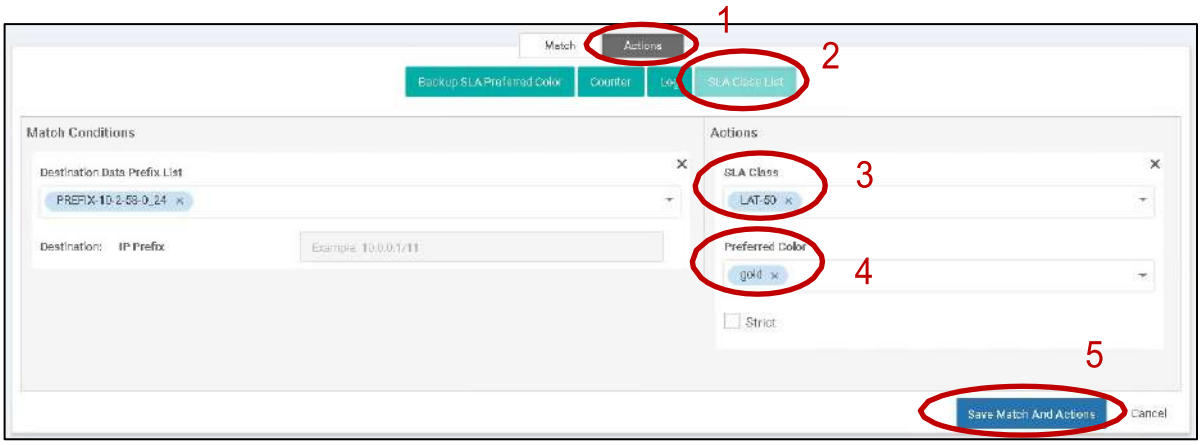
Step 24 Go to the next step.

Step 25 Click **Actions**.

Click **SLA Class List**.

Select **LAT-50** from the **SLA Class** drop-down menu. Select **gold** off the **Preferred Color** drop-down menu. Click **Save Match And Actions** to proceed.

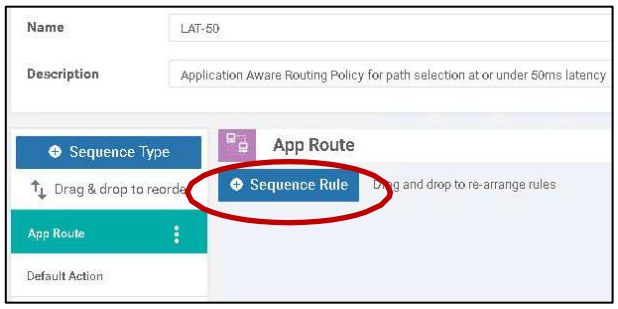
 © 2020 NterOne Corporation



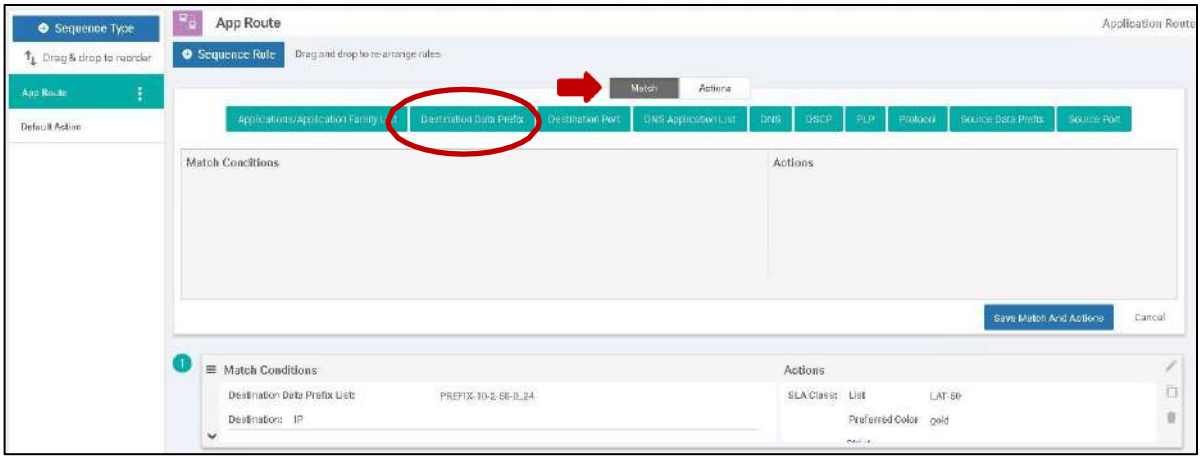
Step 26 You should see the newly added **Sequence Rule** for this App-route policy (arrow).

Step 27 Add a second **Sequence Rule**.

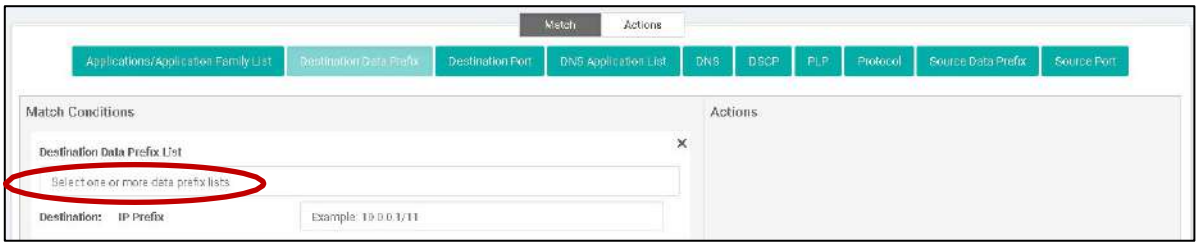
Step 28 Click **Sequence Rule**.



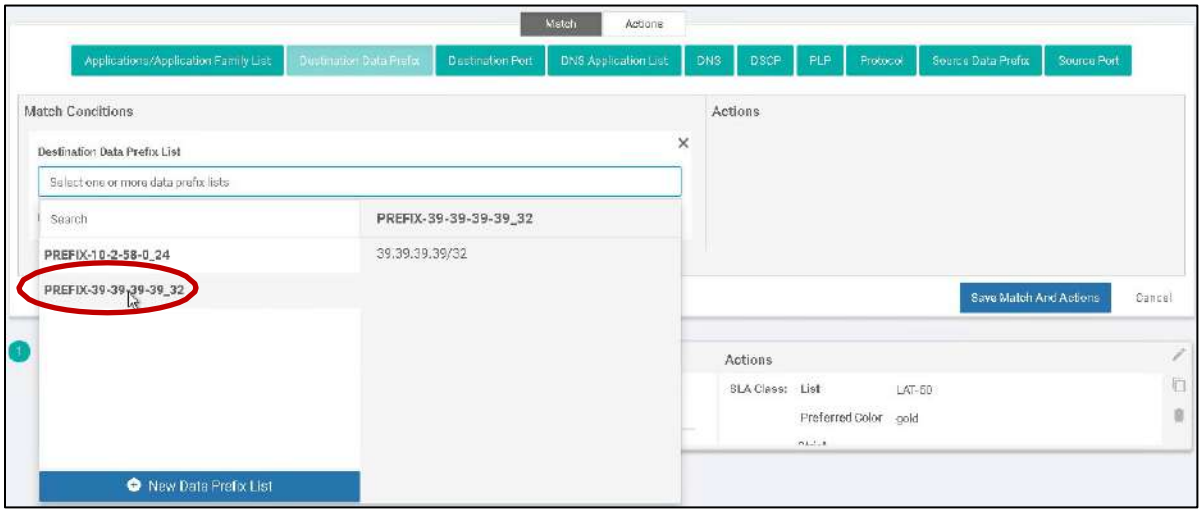
Step 29 Again, the sequence rule should start out with a **Match** (arrow). Click **Destination Data Prefix** from the Match Conditions menu.



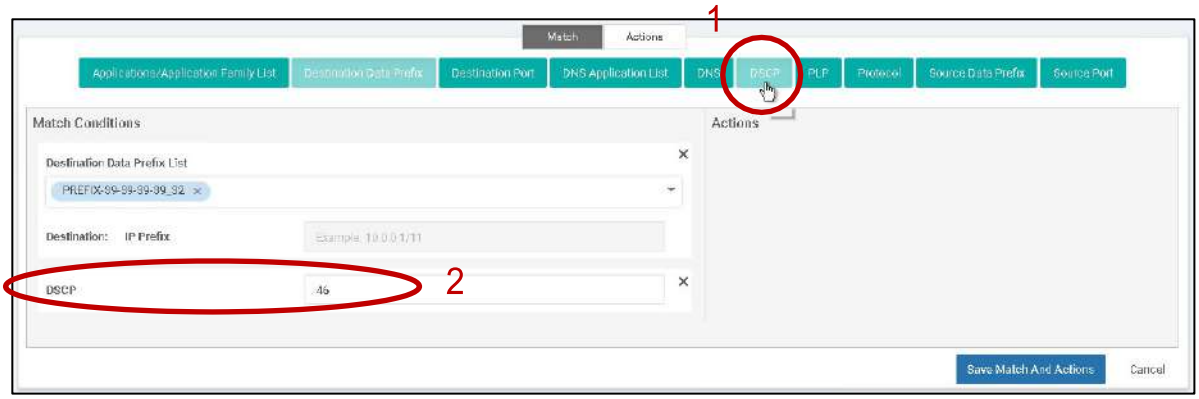
Step 30 Click on the Destination Data Prefix List from the Match Conditions window.



Step 31 Select **PREFIX-39-39-39-39_32** from the Destination Data Prefix List drop-down menu. Do not click **Save Match and Actions yet**



Step 32 You will add one more match condition to this Rule. Click DSCP from the conditions menu. Enter **"46"** for the **DSCP value**.

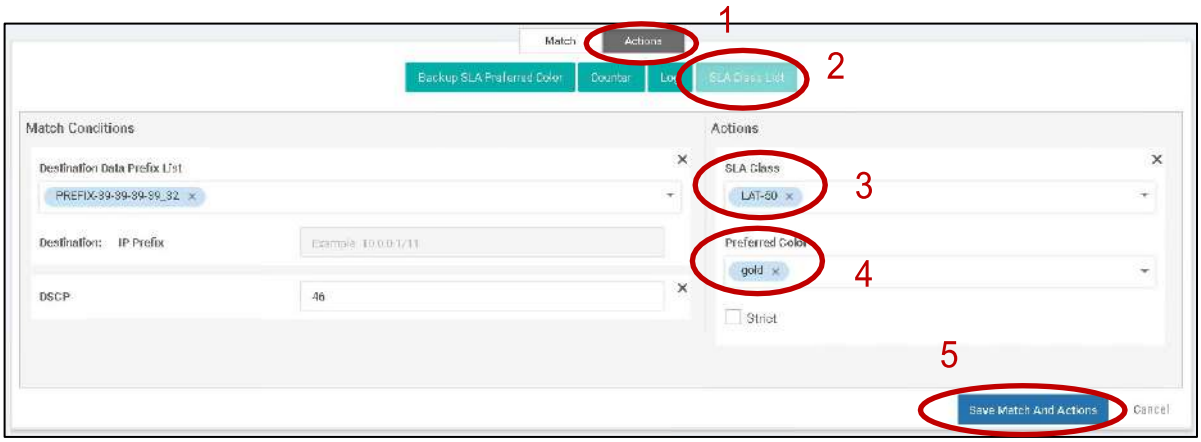


Step 33 Click **Actions**.

Click **SLA Class List**.

Select **LAT-50** from the **SLA Class** drop-down menu. Select **gold** from the **Preferred Color** drop-down menu. Click **Save Match And Actions** to proceed.

 © 2020 NterOne Corporation

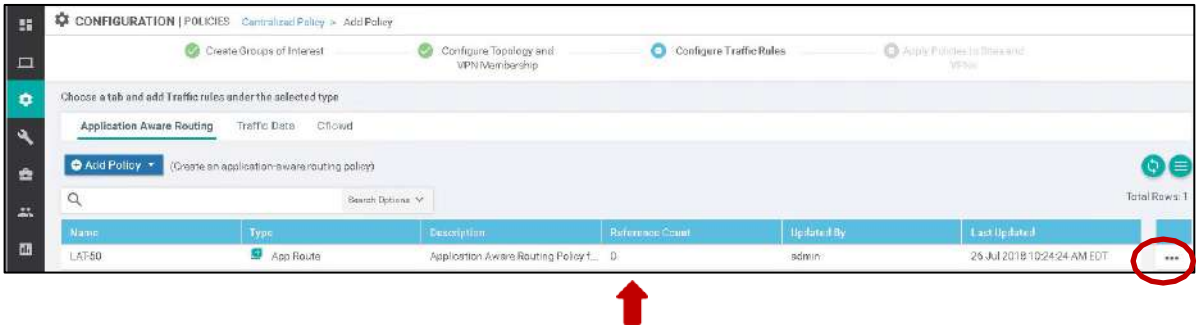


Step 34 You should now have two **Match** Conditions on the list (arrow).

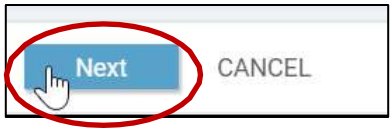
The Default Action does not need to be modified here. (Default = Do nothing) Click on **Save Application Aware Routing Policy.**

Step 35 After you save the AAR Policy, it will be added to the Application Aware Routing Policies list (arrow).

Additional Policies can be added to the list. The Policy can be edited by clicking on **…** and then select edit.



Step 36 Click **Next** to proceed.



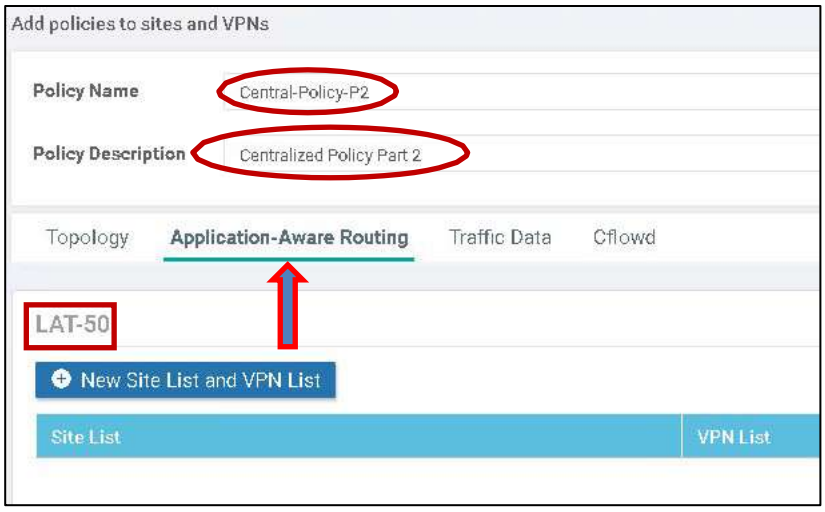
Step 37 This will bring you to the last section - **Apply Policies to Sites and VPNs**.

Here you will **add the Site Lists and or VPN Lists** to which the Policy will be applied.

Step 38 Select **Application-Aware Routing** from the menu (arrow).

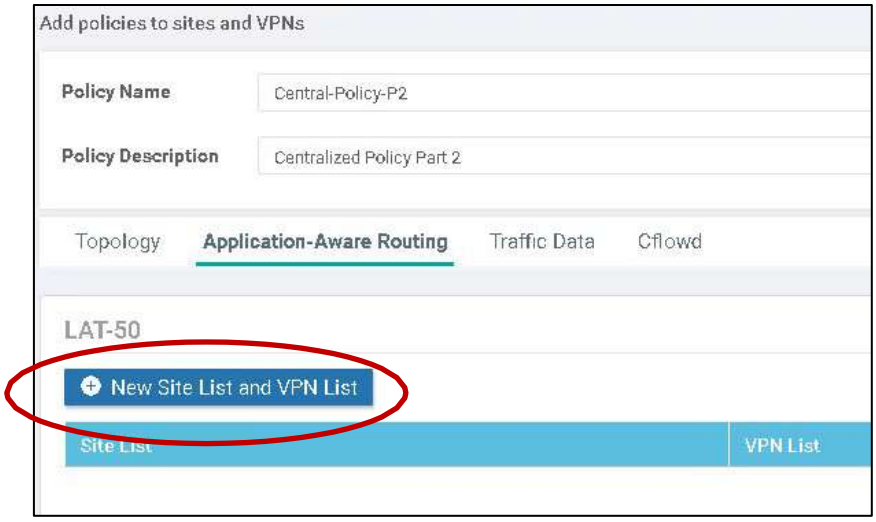
For the **Policy Name** enter **"Central-Policy-P2".** This is the centralized policy name in vManage. Enter a **Description**.

Notice that these settings will apply to the App-route Policy **LAT-50**. If there were any other App-route policies, they would be listed here as well. The same steps could be applied to other App-route policies.

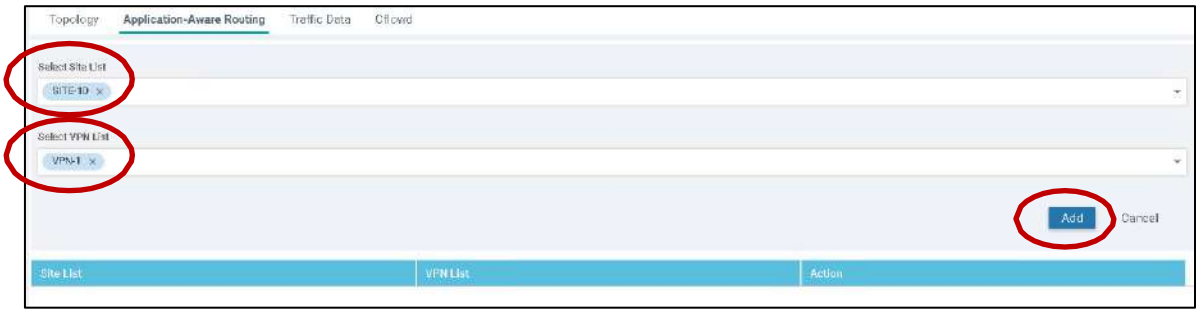


Step 39 Click **+ New Site List and VPN List** to apply the policy to the **Site List and VPN List** created earlier.

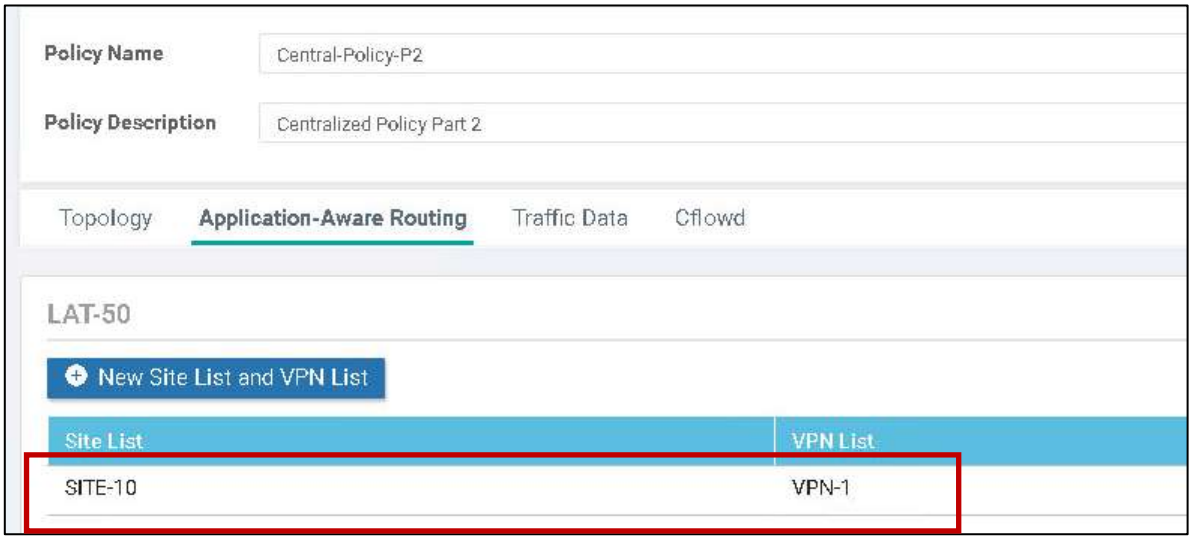
 © 2020 NterOne Corporation



Step 40 Select **SITE-10** from the **Select Site List** drop-down menu. Select **VPN-1** from the **Select VPN List** drop-down menu. Click **Add** when done.



Step 41 You have now applied the **LAT-50** App-route Policy to a **Site List and VPN List**.



Step 42 **(Optional)** If any modifications need to be made to the policy, you can go **BACK** to previous screens (Lower left corner of the window), and then use the **pencil** icon to edit any entries before you save the policy.

(OPTIONAL)

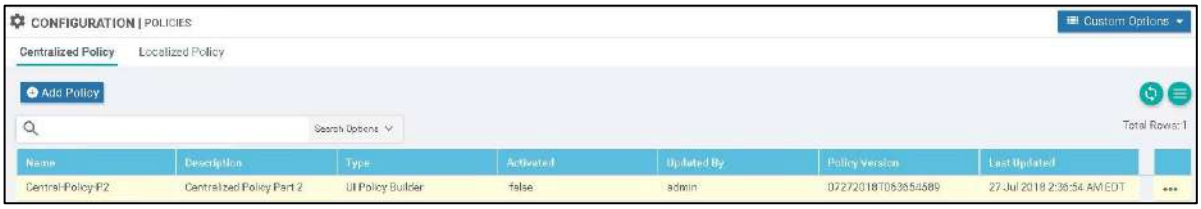
Step 43 Click **Preview** and verify with the output shown below. Go back if you need to edit any entries.

```
  policy
sla-class LAT-50
  latency 50
 !
app-route-policy _VPN-1_LAT-50
 vpn-list VPN-1
   sequence 1
    match
     destination-data-prefix-list PREFIX-10-2-58-0_24
    !
    action
     sla-class LAT-50 preferred color gold
    !
   !
   sequence 11
    match
     destination-data-prefix-list PREFIX-39-39-39-39_32
     dscp 46
    !
    action
     sla-class LAT-50 preferred color gold
    !
   !
!
lists
 data-prefix-list PREFIX-10-2-58-0_24
  ip-prefix 10.2.58.0/24
 !
 data-prefix-list PREFIX-39-39-39-39_32
  ip-prefix 39.39.39.39/32
 !
 site-list SITE-10
  site-id 10
 !
 vpn-list VPN-1
   vpn 1
  !
 !
!
apply-policy
 site-list SITE-10
  app-route-policy _VPN-1_LAT-50
 !
!
```

Step 44 Click **Save Policy**.

 © 2020 NterOne Corporation

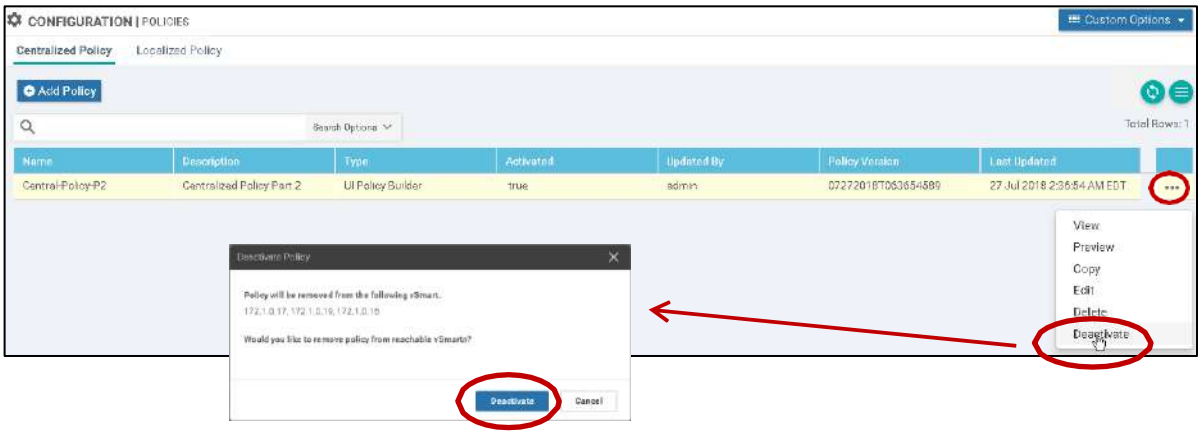
Step 45 You'll be redirected to the **Centralized Policy** summary window.



Step 46 Make sure there is No Active Policy in place on vManage.

Within the Centralized Policy window **deactivate any active policy**.

If there are multiple policies, check each one. Verify with the following example.



Step 47 Go to the CLI of branch-vE1 to verify settings and pathways, BEFORE activating the Policy.

Step 48 Issue the command "show ip route 10.2.58.0/24" to view the next-hop TLOC's.

Note	**Application Aware Routing Policies do not modify the next-hop in the routing table.**

```
branch-vE1# show ip route 10.2.58.0/24
Codes Proto-sub-type:
  IA -> ospf-intra-area, IE -> ospf-inter-area,
  E1 -> ospf-external1, E2 -> ospf-external2,
  N1 -> ospf-nssa-external1, N2 -> ospf-nssa-external2,
  e -> bgp-external, i -> bgp-internal
Codes Status flags:
  F -> fib, S -> selected, I -> inactive,
  B -> blackhole, R -> recursive
```

```
                                          PROTOCOL  NEXTHOP     NEXTHOP          NEXTHOP
VPN    PREFIX              PROTOCOL         SUB TYPE  IF NAME     ADDR             VPN
TLOC IP          COLOR            ENCAP  STATUS
-----------------------------------------------------------------------------------------------
-------------------------------------------------
1      10.2.58.0/24        omp              -         -           -                -
172.1.100.5      gold             ipsec  F,S
1      10.2.58.0/24        omp              -         -           -                -
172.1.200.7      silver           ipsec  F,S
1      10.2.58.0/24        omp              -         -           -                -
172.1.200.8      silver           ipsec  F,S
1      10.2.58.0/24        omp              -         -           -                -
172.1.100.6      gold             ipsec  F,S
```

Step 49 Use the command "show ip fib 10.2.58.0/24" to view the Physical next-hop IP Addresses.

By default **ECMP** will be used for the different flows potentially utilizing all pathways.

```
branch-vE1#  show ip fib 10.2.58.0/24
                        NEXTHOP     NEXTHOP          NEXTHOP  SA
VPN  PREFIX             IF NAME     ADDR             LABEL    INDEX    TLOC IP        COLOR
-----------------------------------------------------------------------------------------------
----------------------------------
1    10.2.58.0/24       ipsec       1.1.15.15        1002     45       172.1.100.5     gold
1    10.2.58.0/24       ipsec       1.1.15.15        1002     46       172.1.100.5     gold
1    10.2.58.0/24       ipsec       1.1.17.16        1002     47       172.1.100.6     gold
1    10.2.58.0/24       ipsec       1.1.17.16        1002     48       172.1.100.6     gold
1    10.2.58.0/24       ipsec       1.2.19.17        1001     53       172.1.200.7     silver
1    10.2.58.0/24       ipsec       1.2.19.17        1001     54       172.1.200.7     silver
1    10.2.58.0/24       ipsec       1.2.21.18        1001     55       172.1.200.8     silver
1    10.2.58.0/24       ipsec       1.2.21.18        1001     56       172.1.200.8     silver
```

Step 50 Use the "show policy from-vsmart" command to view Policies received from vSmart.

```
branch-vE1# show policy from-vsmart
% No entries found.
```

Step 51 **There should be no policies from vSmart**. Otherwise, go back and deactivate the policy from vManage.

Step 52 **Verify the current tunnel path** for a particular **source-destination flow**. Use the following command. "**show policy service-path vpn 1 interface ge0/0 source-ip 10.2.50.19 dest-ip 10.2.58.20 protocol 1**" The **destination prefix** matches the prefix specified in the **LAT-50 App-route Policy.**

 © 2020 NterOne Corporation

```
branch-vE1# show policy service-path vpn 1 interface ge0/0 source-ip 10.2.50.19 dest-ip
10.2.58.20 protocol 1
 Next Hop: IPsec
   Source: 1.2.9.11 12346 Destination: 1.1.17.16 12346 Color: silver (Color may vary)
```

The results should display the **Source Tunnel Physical IP and the Destination Tunnel Physical IP**.

Step 53 **To verify All Tunnels** that can be used for this particular flow, add the **"all"** keyword to the same command.

```
branch-vE1# show policy service-path vpn 1 interface ge0/0 source-ip 10.2.50.19
dest-ip 10.2.58.20 protocol 1 all
Number of possible next hops: 8
Next Hop: IPsec
  Source: 1.1.7.11 12346 Destination: 1.1.15.15 12346 Color: gold
Next Hop: IPsec
  Source: 1.2.9.11 12366 Destination: 1.1.15.15 12346 Color: silver
Next Hop: IPsec
  Source: 1.1.7.11 12346 Destination: 1.1.17.16 12346 Color: gold
Next Hop: IPsec
  Source: 1.2.9.11 12366 Destination: 1.1.17.16 12346 Color: silver
Next Hop: IPsec
  Source: 1.1.7.11 12346 Destination: 1.2.21.18 12346 Color: gold
Next Hop: IPsec
  Source: 1.2.9.11 12366 Destination: 1.2.21.18 12346 Color: silver
Next Hop: IPsec
  Source: 1.1.7.11 12346 Destination: 1.2.19.17 12346 Color: gold
Next Hop: IPsec
  Source: 1.2.9.11 12366 Destination: 1.2.19.17 12346 Color: silver
```

Step 54 It appears that there are **multiple tunnel pathways** available for this flow.

Step 55 **Check the prefix 39.39.39.39** used in the LAT-50 App-route Policy.

```
branch-vE1# show policy service-path vpn 1 interface ge0/0 source-ip
10.2.50.19 dest-ip 39.39.39.39 protocol 1
 Next Hop: IPsec
   Source: 1.1.7.11 12346 Destination: 1.1.15.15 12346 Color: gold
```

Step 56 Check the prefix again, but this time, add "**dscp 46**".

```
branch-vE1# show policy service-path vpn 1 interface ge0/0 source-ip
10.2.50.19 dest-ip 39.39.39.39 protocol 1 dscp 46
 Next Hop: IPsec
   Source: 1.2.9.11 12346 Destination: 1.1.17.16 12346 Color: silver
```

Step 57 There's no path difference for the **prefix 39.39.39.39 when used with the DSCP value 46. This will change once the policy has been activated**.

Step 58 Check the tunnel statistics with the command **"show app-route stats"**.

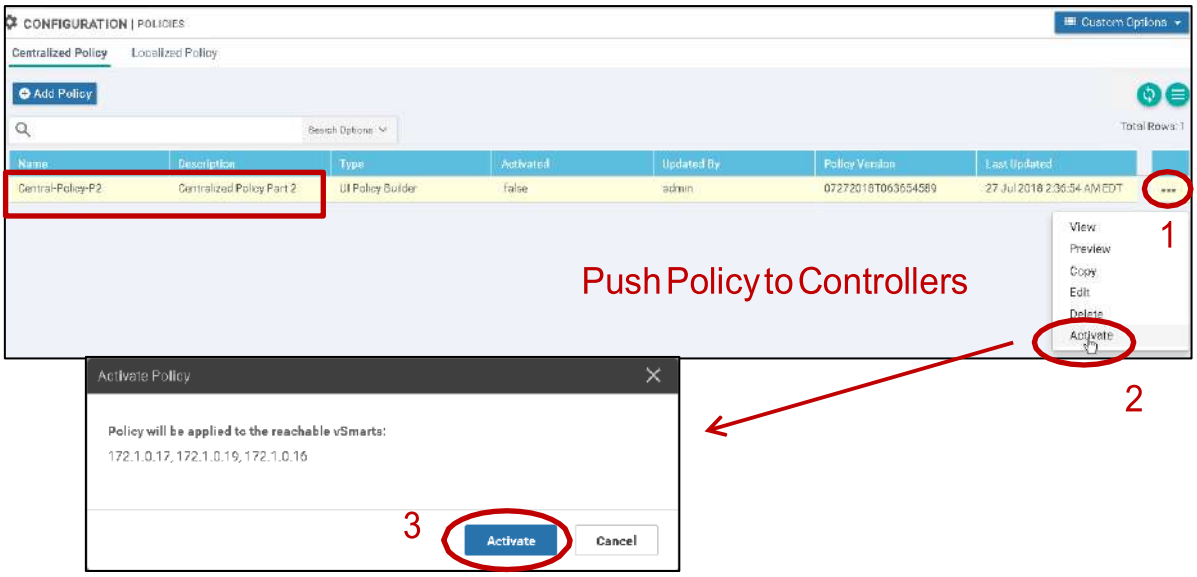
```
branch-vE1# show app-route stats
 app-route statistics 1.1.7.11 1.1.10.12 ipsec 12346 12346
  remote-system-ip 172.1.20.2
  local-color      gold
  remote-color     gold
  mean-loss        5
  mean-latency     99
  mean-jitter      1
  sla-class-index  0

        TOTAL          AVERAGE  AVERAGE  TX DATA  RX DATA
 INDEX  PACKETS  LOSS  LATENCY  JITTER   PKTS     PKTS
 ----------------------------------------------------------
 0      600      31    99       0        0        0
 1      600      35    99       0        0        0
 2      600      32    99       1        0        0
 3      600      38    99       1        0        0
 4      600      23    99       0        0        0
 5      600      26    99       1        0        0

 app-route statistics 1.2.9.11 1.1.10.12 ipsec 12346 12346
  remote-system-ip 172.1.20.2
  local-color      silver
  remote-color     gold
  mean-loss        0
  mean-latency     0
  mean-jitter      0
  sla-class-index  0

        TOTAL          AVERAGE  AVERAGE  TX DATA  RX DATA
 INDEX  PACKETS  LOSS  LATENCY  JITTER   PKTS     PKTS
 ----------------------------------------------------------
 0      600      0     0        0        0        0
 1      600      0     0        0        0        0
 2      600      0     0        0        0        0
 3      600      0     0        0        0        0
 4      600      0     0        0        0        0
```

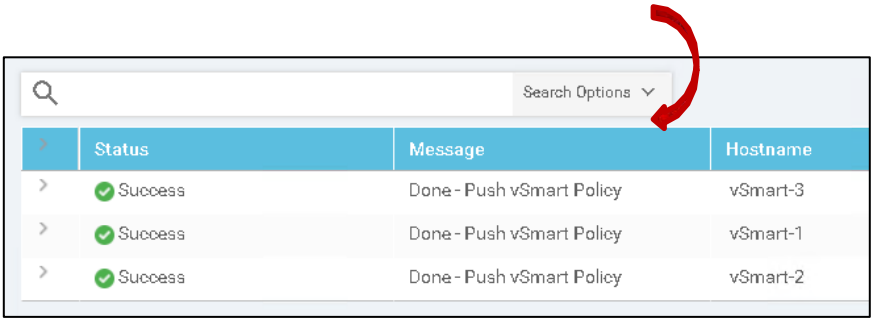
 © 2020 NterOne Corporation

```
  5        600        0        0          0          0          0
--- OUTPUT OMITTED --
```

Step 59 Notice that the latency is **~100ms for all gold sourced tunnels and 1-3 ms for silver.**

Step 60 **Go to vManage - Configuration → Policies and Activate the centralized policy Central-Policy-P2**





Step 61 You'll be redirected to **Task View to confirm the Policy Push**. Contact your Instructor if anything fails.

Step 62 Go back to branch-vE1 and once again verify policies received from vSmart.

Note	**Contact your Instructor if you are not receiving any policy.**

```
branch-vE1# show policy from-vsmart
 from-vsmart sla-class LAT-50
  latency 50
 from-vsmart app-route-policy _VPN1_LAT-50
```

```
 vpn-list VPN1
  sequence 1
   match
    destination-data-prefix-
list PREFIX-10-2-58-0_24
   action
    sla-class       LAT-50
    no sla-class strict
    sla-class preferred-color
gold
  sequence 11
   match
    destination-data-prefix-
list PREFIX-39-39-39-39_32
    dscp
46
   action
    sla-class       LAT-50
    no sla-class strict
    sla-class preferred-color
gold
from-vsmart lists vpn-list
VPN1
 vpn 1
from-vsmart lists data-prefix-
list PREFIX-10-2-58-0_24
 ip-prefix 10.2.58.0/24
from-vsmart lists data-prefix-
list PREFIX-39-39-39-39_32
 ip-prefix 39.39.39.39/32
```

Step 63 **Verify the prefixes again and notice the changes to the tunnel pathways.**

Step 64 **Verify the current tunnel path for the destination 10.2.58.20**. Use the following command. **"show policy service-path vpn 1 interface ge0/0 source-ip 10.2.50.19 dest-ip 10.2.58.20 protocol 1"**

```
branch-vE1# show policy service-path vpn 1 interface ge0/0 source-ip
10.2.50.19 dest-ip 10.2.58.20 protocol 1
Next Hop: IPsec
  Source: 1.1.7.11 12346 Destination: 1.1.15.15 12346 Color: silver
```

Step 65 **The destination 10.2.58.20** is now using the **silver tunnel path**.

Step 66 Once again, verify **which tunnels are available for this flow** by using the **"all"** keyword. "show policy service-path vpn 1 interface ge0/0 source-ip 10.2.50.19 dest-ip 10.2.58.20 protocol 1 all"

 © 2020 NterOne Corporation

Output displayed on the next page.

```
branch-vE1# show policy service-path vpn 1 interface ge0/0 source-ip
10.2.50.19 dest-ip 10.2.58.20 protocol 1 all
Number of possible next hops: 4
Next Hop: IPsec
  Source:    1.2.9.11 12346 Destination: 1.1.15.15 12346    Color: silver
Next Hop:    IPsec
     Source: 1.2.9.11 12346 Destination: 1.1.17.16 12346    Color: silver
   Next Hop: IPsec
     Source: 1.2.9.11 12346 Destination: 1.2.19.17 12346    Color: silver
   Next Hop: IPsec
     Source: 1.2.9.11 12346 Destination: 1.2.21.18 12346    Color: silver
```

Step 67 **The only available tunnels** for this destination appear to be **Silver** due to the **applied App-route policy.**

Step 68 **Verify the prefix 39.39.39.39** with and without the DSCP value. **First, verify without the value.**

```
branch-vE1# show policy service-path vpn 1 interface ge0/0
source-ip 10.2.50.19 dest-ip 39.39.39.39 protocol 1
Next Hop: IPsec
  Source: 1.1.7.11 12346 Destination: 1.1.15.15 12346 Color: gold
```

Step 69 The tunnel might not be silver, which is fine. Verify which tunnels are available for this destination.

Make sure to use the **"all"** keyword.

```
branch-vE1# show policy service-path vpn 1 interface ge0/0
source-ip 10.2.50.19 dest-ip 39.39.39.39 protocol 1 all
Number of possible next hops: 4
Next Hop: IPsec
  Source:   1.1.7.11 12346 Destination: 1.1.15.15 12346   Color: gold
Next Hop:   IPsec
    Source: 1.2.9.11 12346 Destination: 1.1.15.15 12346   Color: silver
  Next Hop: IPsec
    Source: 1.1.7.11 12346 Destination: 1.1.17.16 12346   Color: gold
  Next Hop: IPsec
    Source: 1.2.9.11 12346 Destination: 1.1.17.16 12346   Color: silver
```

Step 70 It appears that without the DSCP value, **all tunnels are available** for the **destination 39.39.39.39**. This is due to the fact that our policy is looking for both the **destination AND the DSCP value 46**.

Step 71 Let's try the command again. **This time use DSCP value 46**.

```
branch-vE1# show policy service-path vpn 1 interface ge0/0
source-ip 10.2.50.19 dest-ip 39.39.39.39 protocol 1 dscp 46
Next Hop: IPsec
  Source: 1.2.9.11 12346 Destination: 1.1.15.15 12346 Color: silver
```

Step 72 With the App-route policy in place, the Silver tunnel should be selected due to the higher latency of the gold tunnel.

Step 73 Once again, check which tunnels are available with the **"all"** keyword.

```
branch-vE1# show policy service-path vpn 1 interface ge0/0
source-ip 10.2.50.19 dest-ip 39.39.39.39 protocol 1 dscp 46 all
Number of possible next hops: 2
Next Hop: IPsec
  Source: 1.2.9.11 12346 Destination: 1.1.15.15 12346 Color: silver
Next Hop: IPsec
  Source: 1.2.9.11 12346 Destination: 1.1.17.16 12346 Color: silver
```

Step 74 **Our policy prefers pathways with a latency of 50ms or lower. Since all of the gold sourced traffic is above 50ms on average, only the silver tunnels remains as an option.**

Step 75 **Let's confirm the tunnel preference for "gold" which was configured in the LAT-50 Policy.**

Step 76 Even though our **policy prefers gold for the tunnel preference,** it's overridden due to the **50ms latency requirement** within the **SLA Class** specified in the policy.

Step 77 **Test the gold tunnel preference** in the policy. For this to work, we will need to make some **adjustments to the Policy.**

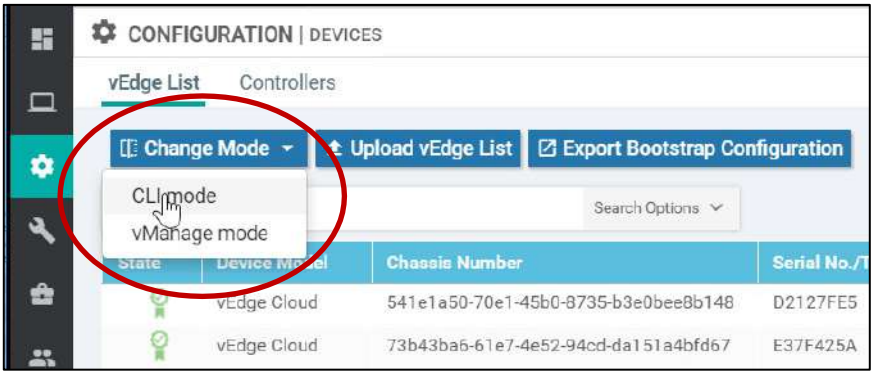
Step 78 **Although not necessary for Preference testing, adjust the BFD Multiplier and Polling Interval. This will be done in Step 86.**

Step 79 First**, remove any associated template** from the vEdge router, so that we can adjust it manually.

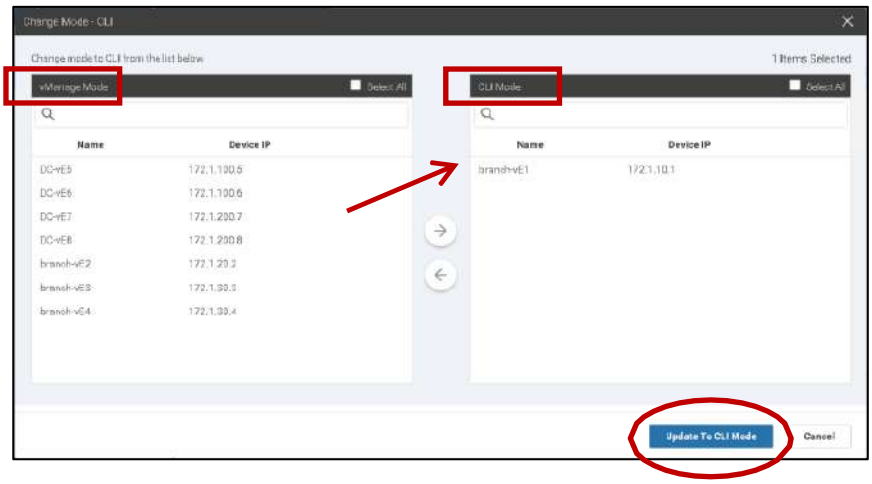
Step 80 In vManage go to **Configuration > Devices** and highlight the **branch-vE1** router**.**

Step 81 Make sure that **branch-vE1 is still selected,** then go to the top left corner of the window and click **Change Mode**, and then select **CLI mode.** (This will remove the Template from the device)

 © 2020 NterOne Corporation



Step 82 The **Change Mode - CLI** window will appear. Select **branch-vE1** from the **vManage Mode** list and add it to the right under **CLI Mode**. Click **Update To CLI Mode** when done.



Step 83 Confirm from the **Configuration → Devices list that it has been changed to CLI mode**. Contact your instructor if you encounter any problems.

Device Model	Chassis Number	Serial No./Token	Hostname	State	Site ID	Mode
vEdge Cloud	2e836293-647f-43a7-8323-0cad25e2d6...	d35cfdb3	branch-vE1		10	CLI

Step 84 Verify what the **current Polling Interval and Multiplier** value is. Use the following commands.

```
branch-vE1# show bfd summary poll-interval
bfd summary poll-interval 600000

branch-vE1# show app-route stats
app-route statistics 1.1.7.11 1.1.10.12 ipsec 12346 12346
remote-system-ip 172.1.20.2
local-color      gold
remote-color     gold
```

```
mean-loss        4
mean-latency     99
mean-jitter      1
sla-class-index 0


        TOTAL           AVERAGE  AVERAGE  TX DATA  RX DATA

INDEX   PACKETS  LOSS   LATENCY  JITTER   PKTS     PKTS
-------------------------------------------------------
0       600      30     99       0        0        0
1       600      17     99       1        0        0
2       600      28     99       1        0        0
3       600      34     99       0        0        0
4       600      36     99       0        0        0
5       600      27     99       1        0        0

--- OUTPUT OMITTED ---
```

Step 85 It appears that the **Polling Interval** is set to the **default 600,000 milliseconds** (10 minutes).

The **Multiplier** (Index) seems to be set to **6 by default** (0 – 5)

Step 86 Change the **Polling Interval to 1 minute** and the **Multiplier to 2**. Use the following commands.

```
branch-vE1# conf
Entering configuration mode terminal
branch-vE1(config)# bfd app-route poll-interval 60000
branch-vE1(config)# bfd app-route multiplier 2
branch-vE1(config)# end
Uncommitted changes found, commit them? [yes/no/CANCEL] yes
Commit complete.
```

Step 87 Wait a few minutes, and then confirm the configuration changes that were made.

```
branch-vE1# show bfd summary poll-interval
bfd summary poll-interval 60000


branch-vE1# show app-route stats
app-route statistics 1.1.7.11 1.1.10.12 ipsec 12346 12346
remote-system-ip 172.1.20.2
local-color      gold
```

 © 2020 NterOne Corporation

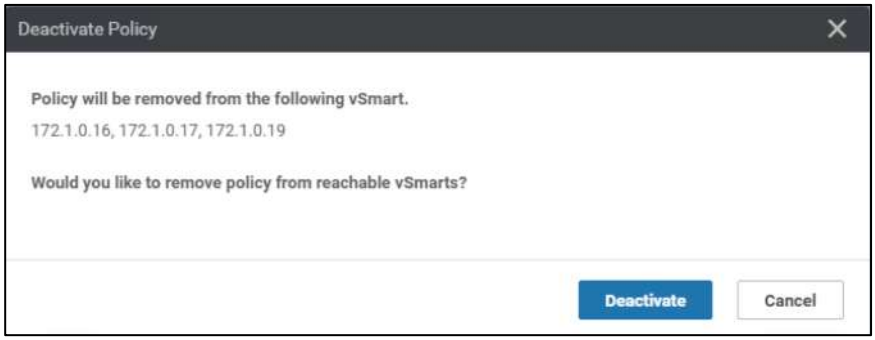
```
remote-color     gold
mean-loss 3
mean-latency     99
mean-jitter      2
sla-class-index 0

       TOTAL          AVERAGE  AVERAGE  TX DATA  RX DATA
INDEX  PACKETS  LOSS  LATENCY  JITTER   PKTS     PKTS
-----------------------------------------------------------
0      60       3     100      3        0        0
1      60       1     99       0        0        0

app-route statistics 1.2.9.11 1.1.10.12 ipsec 12346 12346
remote-system-ip 172.1.20.2 local-color    silver
remote-color     gold
mean-loss 0
mean-latency     0
mean-jitter      0
sla-class-index 0,1

       TOTAL          AVERAGE  AVERAGE  TX DATA  RX DATA
INDEX  PACKETS  LOSS  LATENCY  JITTER   PKTS     PKTS
-----------------------------------------------------------
0      60       0     0        0        0        0
1      60       0     0        0        0        0

--- OUTPUT OMITTED ---
```

Step 88 Confirm that the tunnel used for the prefix 10.2.58.20 is still silver. Use the following command.

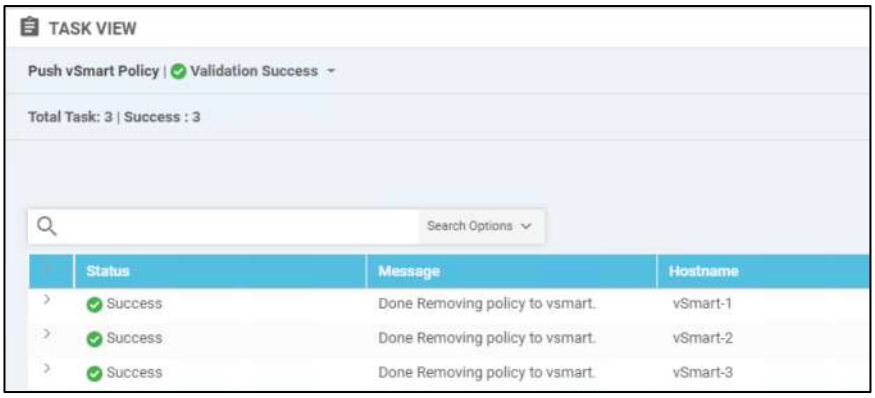
```
branch-vE1# show policy service-path vpn 1 interface ge0/0 source-ip
10.2.50.19 dest-ip 10.2.58.20 protocol 1 all
Number of possible next hops: 4 Next Hop: IPsec
  Source: 1.2.9.11 12346 Destination: 1.1.15.15 12346  Color: silver
Next Hop: IPsec
  Source: 1.2.9.11 12346 Destination: 1.1.17.16 12346  Color: silver
     Next IPsec
     Hop:
  Source: 1.2.9.11 12346 Destination: 1.2.19.17 12346  Color: silver
     Next IPsec
     Hop:
  Source: 1.2.9.11 12346 Destination: 1.2.21.18 12346  Color: silver
```

Step 89 **The LAT-50 policy needs to be adjusted** so that the latency requirements are higher than the configured 50ms value.

Step 90 Navigate to **Configuration > Policies**, select **Central-Policy-32**, then click **…** and choose **Deactivate** from the pull-down menu. Click the **Deactivate** button from the Deactivate Policy pop-up menu to confirm.



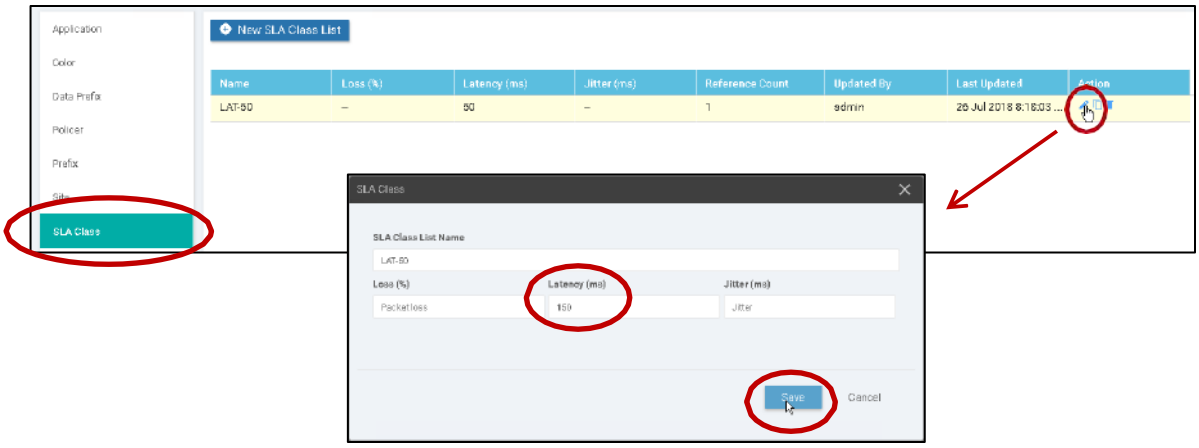
Step 91 Wait for the Task View to show Status "**Success**" for the three vSmart devices.



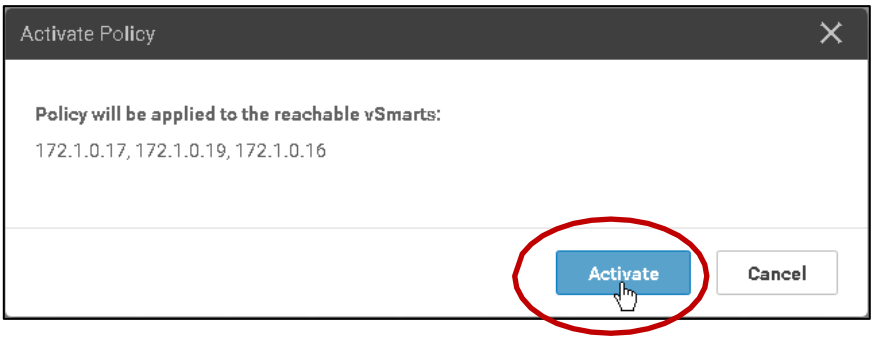
Step 92 In vManage go to **Configuration > Policies** and **edit the LAT-50 SLA Class.** To do this click on **Custom Options** located at the top-right corner of the policies window (arrow). Select **Lists** from the drop-down menu.

 © 2020 NterOne Corporation

Step 93 Select **SLA Class** from the List Type field, and then **click the pencil icon** from the **Actions column** of the **LAT-50** SLA Class. Change the **Latency to 150ms**, and then click **Save.**



Step 94 **Activate** the Policy and then confirm successful completion. **A modified policy will first need to be de-activated before it can be activated.**



Step 95 **Go back to branch-vE1** and verify that the Latency has been changed. Use the following command.

```
branch-vE1# show app-route sla-class

INDEX    NAME                    LOSS   LATENCY   JITTER
-----------------------------------------------------

0        __all_tunnels___        0      0         0
1        LAT-50                  0      150       0
```

Step 96 Verify which tunnel path is being used by the **destination 10.2.58.20.** Use the following commands.

```
branch-vE1# show policy service-path vpn 1 interface ge0/0 source-ip
10.2.50.19 dest-ip 10.2.58.20 protocol 1
Next Hop: IPsec
  Source: 1.1.7.11 12346 Destination: 1.1.15.15 12346 Color: gold
```

```
branch-vE1# show policy service-path vpn 1 interface ge0/0 source-ip
10.2.50.19 dest-ip 10.2.58.20 protocol 1 all
Number of possible next hops: 4
Next Hop: IPsec
  Source: 1.1.7.11 12346 Destination: 1.1.15.15 12346 Color: gold
Next Hop: IPsec
  Source: 1.1.7.11 12346 Destination: 1.1.17.16 12346 Color: gold
Next Hop: IPsec
  Source: 1.1.7.11 12346 Destination: 1.2.21.18 12346 Color: gold
Next Hop: IPsec
  Source: 1.1.7.11 12346 Destination: 1.2.19.17 12346 Color: gold
```

Step 97 Now that the latency is not the issue, the **Preferred Tunnel "gold"** is used.

 © 2020 NterOne Corporation

Task 2: Create a Service Chain Policy

You will create a Service Chain Policy by editing an existing Centralized Policy.

Activity Procedure

Complete these steps:

Note	THIS TASK REQUIRES COMPLETION of the PREVIOUS TASK "Application Aware Routing Policy"

Step 1 **First, verify the settings BEFORE** creating the Service Chain Policy.

Step 2 On branch-vE1 issue the command "show ip route 10.2.54.0/24" to view the next-hop TLOC's. Service Chaining Policies (control based) will modify the next-hop in the routing table.

```
branch-vE1# show ip route 10.2.54.0/24                    (Formatted for clarity)

--- LEGEND OMITTED ---

PROTOCOL  NEXTHOP NEXTHOP NEXTHOP
VPN  PREFIX        PROTOCOL   SUB TYPE   IF NAME ADDR   VPN TLOC IP     COLOR  ENC  STAT
-----------------------------------------------------------------------------------------
1    10.2.54.0/24  omp        -          -              -  172.1.100.5 gold   ipsec F,S
```

Step 3 Use the command "**show ip fib 10.2.54.0/24**" to view the **Physical next-hop IP Addresses.**

```
branch-vE1# show ip fib 10.2.54.0/24     (Formatted for clarity)

                      NEXTHOP   NEXTHOP      NEXTHOP   SA
VPN    PREFIX         IF NAME   ADDR         LABEL     INDEX     TLOC IP          COLOR
-----------------------------------------------------------------------------------------
1      10.2.54.0/24   ipsec     1.1.15.15    1002      179       172.1.100.5      gold
1      10.2.54.0/24   ipsec     1.1.15.15    1002      180       172.1.100.5      gold
```

Step 4 **Currently all traffic destined for 10.2.54.0/24** is being forwarded to the **next-hop 172.1.100.5**

Step 5 **Go to vSmart-1** and verify if any **Control policies** are present.

```
vSmart-1# show running-config policy control-policy
% No entries found
```

Step 6 View all **policy values** using the following commands.

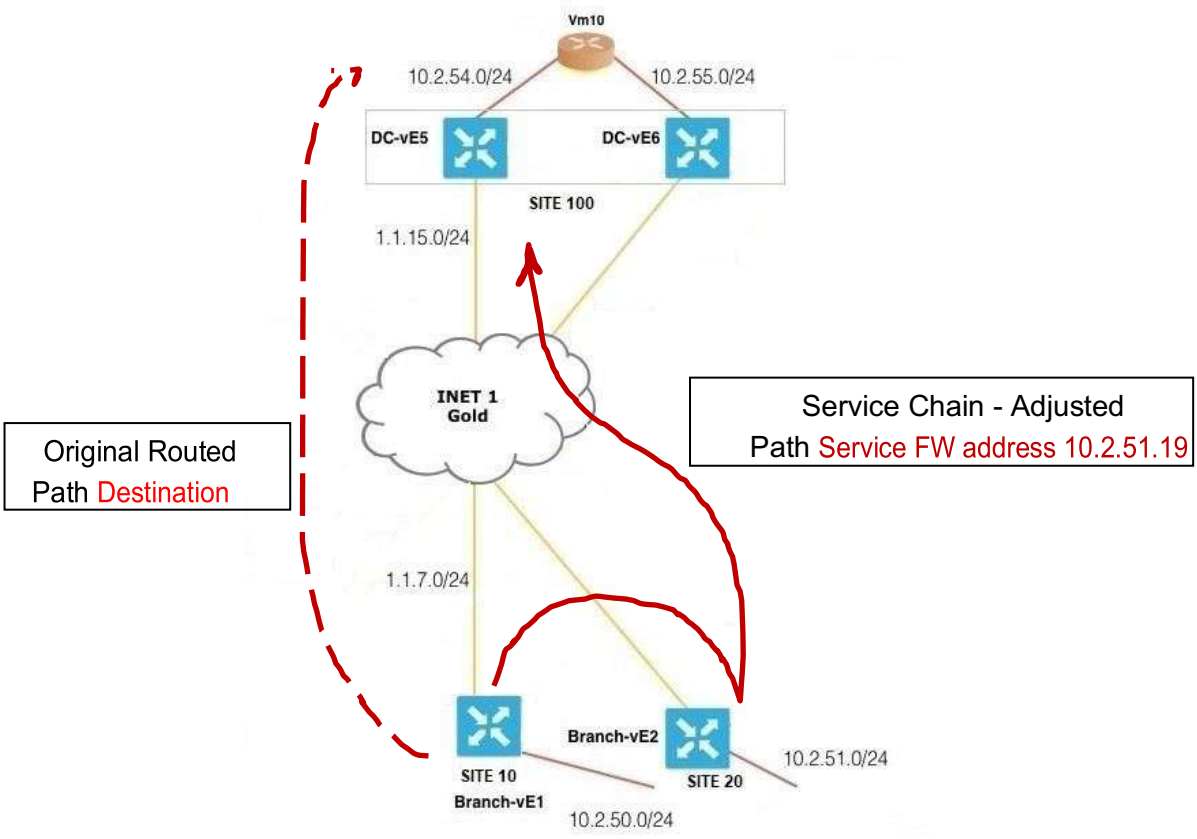
```
vSmart-1# show run policy
policy
 sla-class LAT-50
  latency 150
 !
 app-route-policy _VPN-1_LAT-50
--- OUTPUT OMITTED ---
vSmart-1# show run apply-policy
apply-policy
 site-list SITE-10
  app-route-policy _VPN-1_LAT-50
 !
!
```

Step 7 Currently, **the only policy that is in place and applied** is the **App-route policy created previously**. Leave the **App-route policy in place** and go to the next step.

Step 8 In the next few steps, you will **create a Service Chaining Policy** using the following criteria.

- Destination: **10.2.54.0/24**
- Service: **Firewall**
- Service IP: **10.2.51.19**
- Service TLOC: **172.1.20.2**
- Service VPN: **1**
- Service color: **gold**

 © 2020 NterOne Corporation

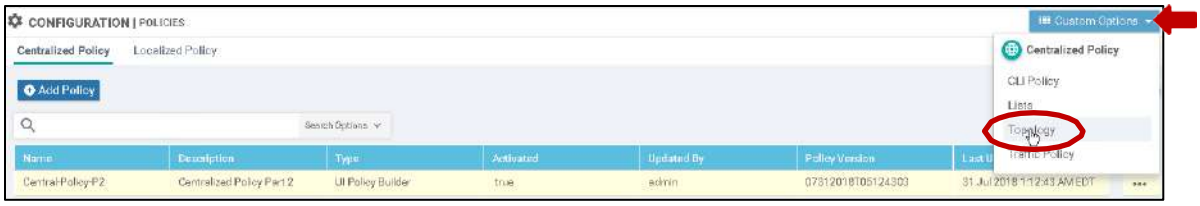


Step 9 Open a browser window and **login to vManage**. You will use vManage to create Policies.

Step 10 You will create a **Service Chain Policy** that will redirect data towards a specific destination/service. You will use a **Control Policy** that will allow this **Service Chain to be redirected to a FW service**.

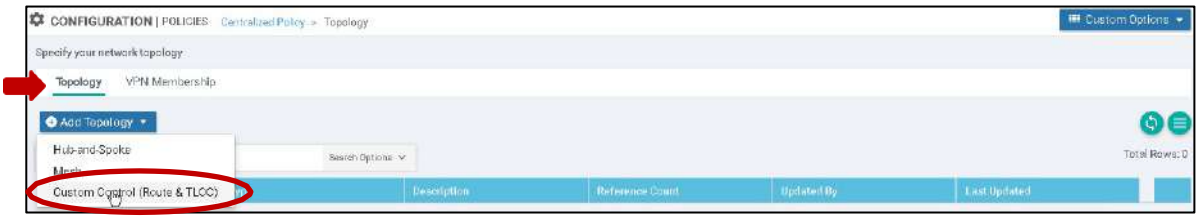
Step 11 In vManage go to **Configuration > Policies > Custom Options** and select **Topology**.

To do this click on **Custom Options** located at the top-right corner of the policies window (arrow). Select **Topology** from the drop-down menu.



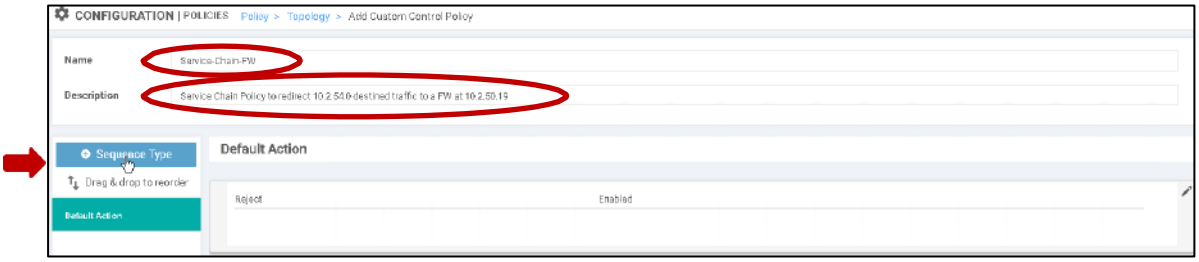
Step 12 Select **Topology** from the menu (default).

Click **+ Add Topology** and select **Custom Control (Route & TLOC)** from the drop-down menu.

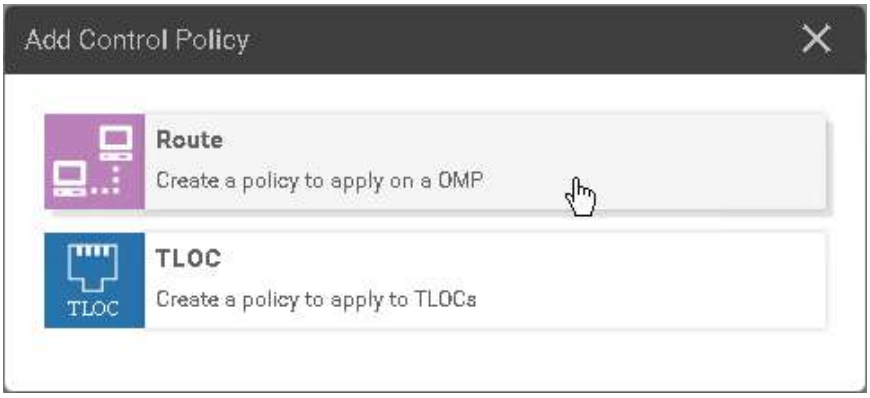


Step 13 Use **"Service-Chain-FW"** for the Policy Name and then provide a **Description**.

Click on **+ Sequence Type** (arrow).



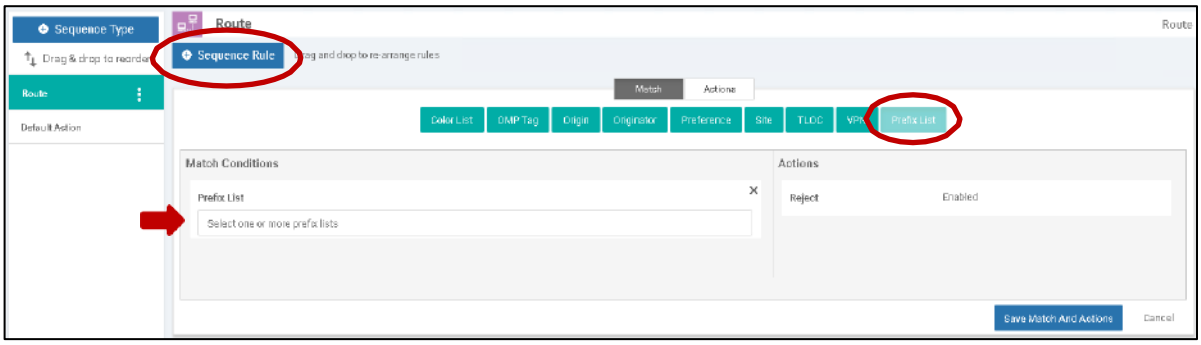
Step 14 Select Route from the Add Control Policy options.



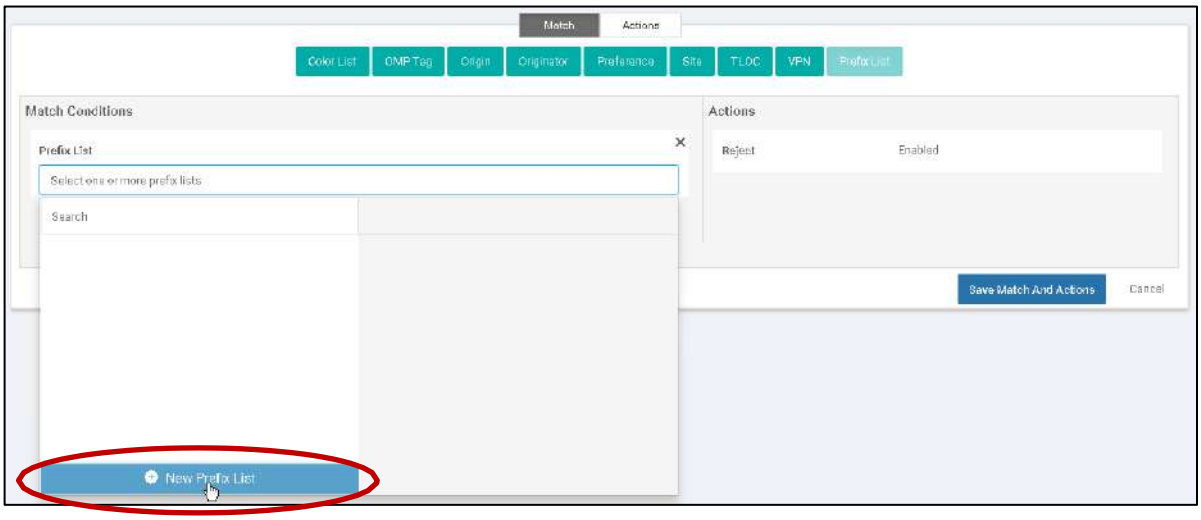
Step 15 Select **+ Sequence Rule** from the Route window.

 © 2020 NterOne Corporation

Select **Prefix List** from the Match Conditions menu. Click on **Select one or more prefix lists** (arrow).



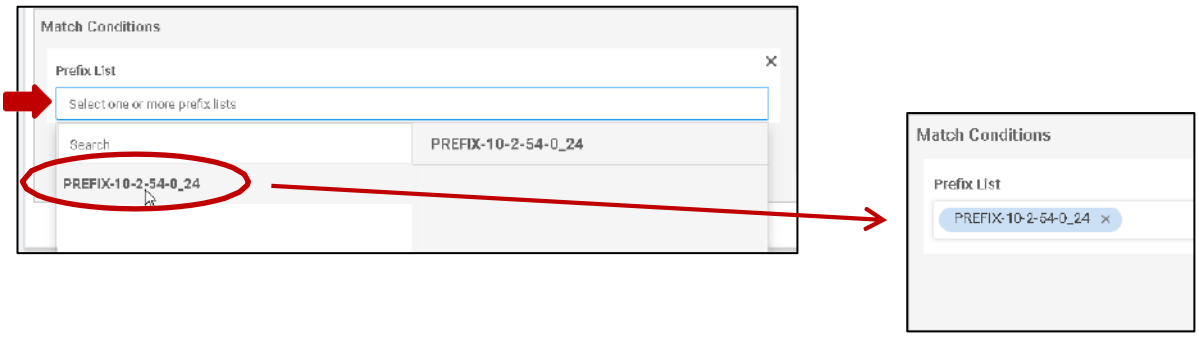
Step 16 Once the drop-down menu appears click **+ New Prefix List.** You'll create a new list from here.



Step 17 Enter **"PREFIX-10-2-54-0_24"** for the Prefix List Name.

Use **"10.2.54.0/24"** for the Add Prefix. Click **Save** when done.

Step 18 Again, click the drop-down list (arrow) and select the newly created list **PREFIX-10-2-54-0_24.**

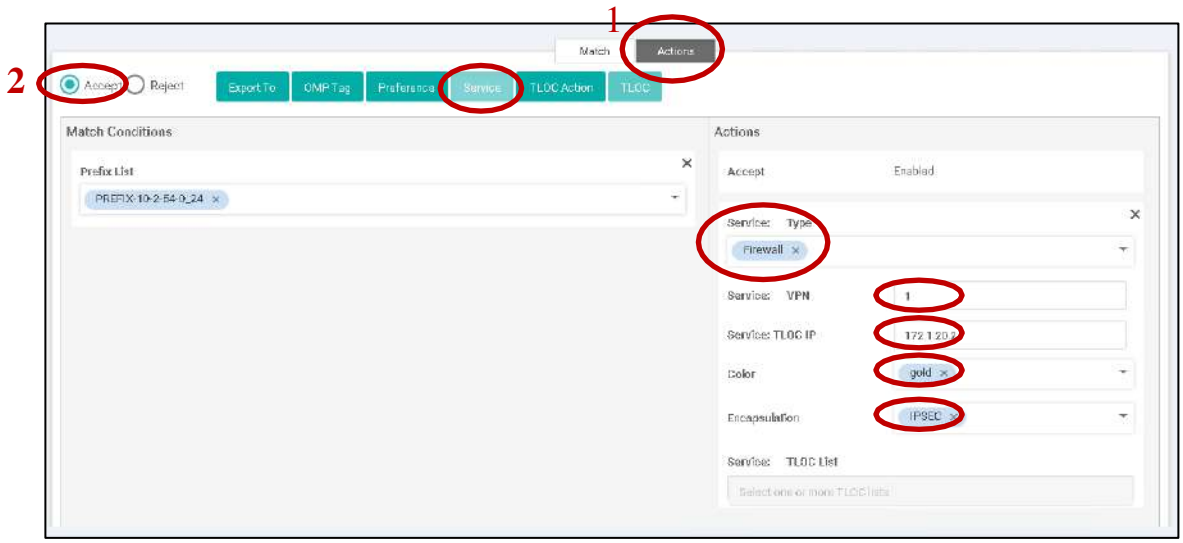


Step 19 Click on **Actions** and then the **Accept** radio button**.** (1 and 2 below)

Select **Service** from the actions menu.

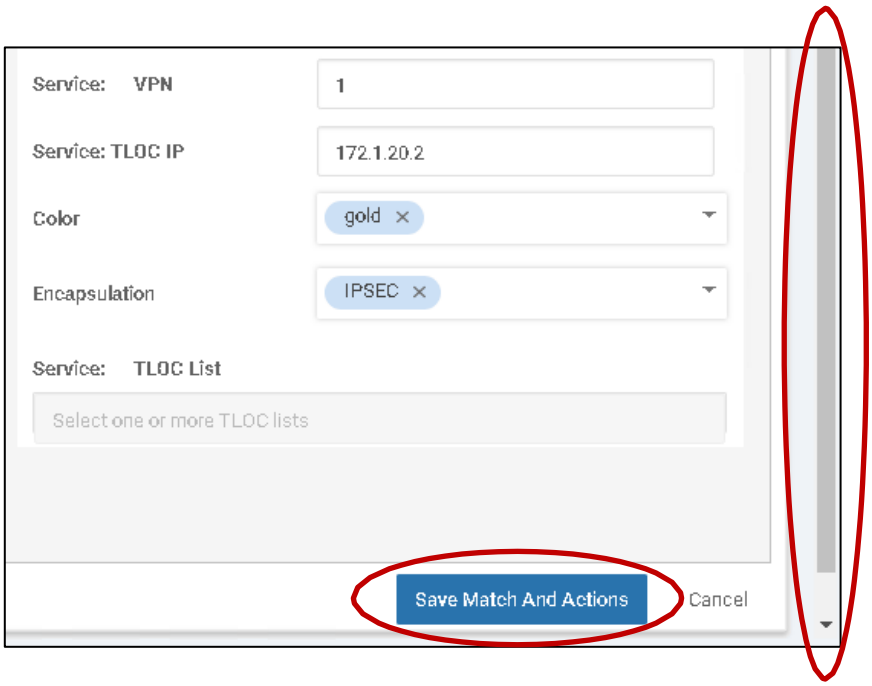
Over to the right side on the Actions window select or type the following:

- Service: Type **Firewall**
- Service: VPN **1**
- Service: TLOC IP **172.1.20.2**
- Color **gold**
- Encapsulation **IPSEC**



Step 20 Over to the right side, scroll down and click **Save Match And Actions**.

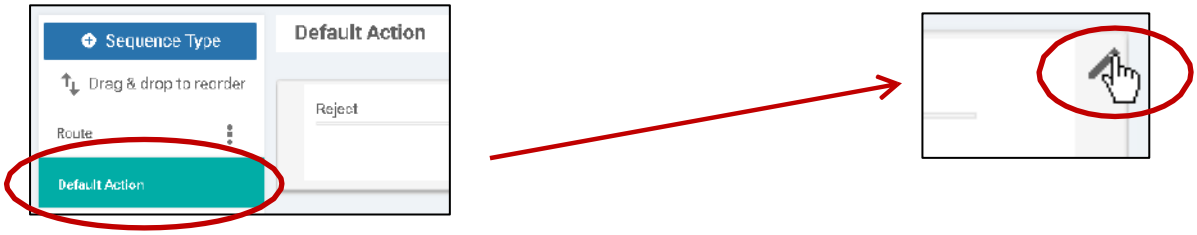
 © 2020 NterOne Corporation



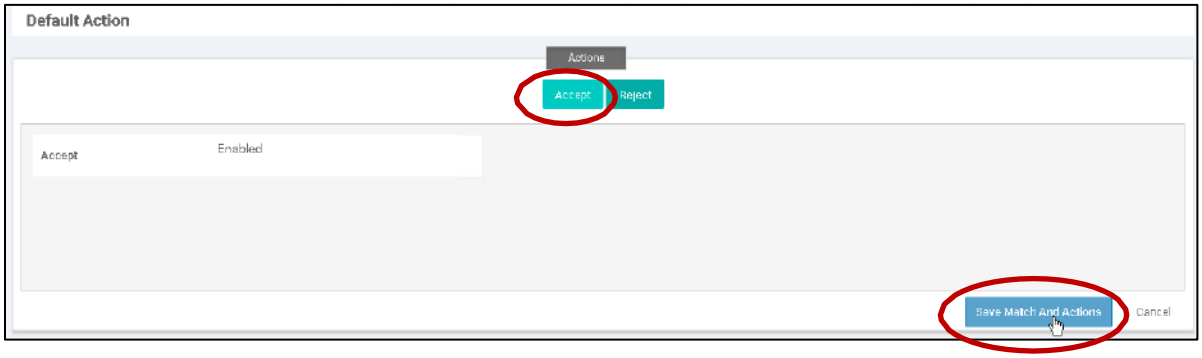
Step 21 Only one Sequence Rule is required.

Step 22 The Default Action needs to be modified to allow all remaining traffic to be forwarded without any modification. Select **Default Action** from the left side of the window.

Click on the **pencil icon** over to the right to edit the default action.



Step 23 Click **Accept**, and then select **Save Match And Actions** to continue.

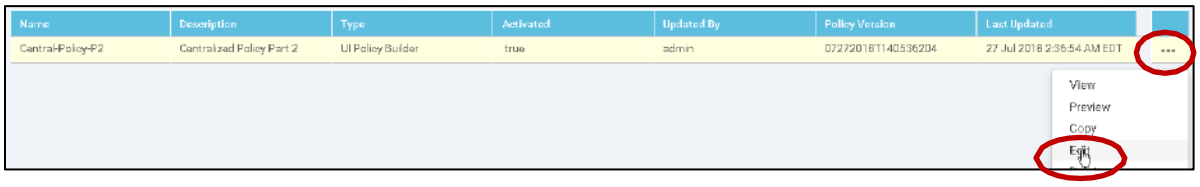


Step 24 Go back and select "**Service-Chain-FW**"

Step 25 Click **Save Control Policy** and then **Activate**.

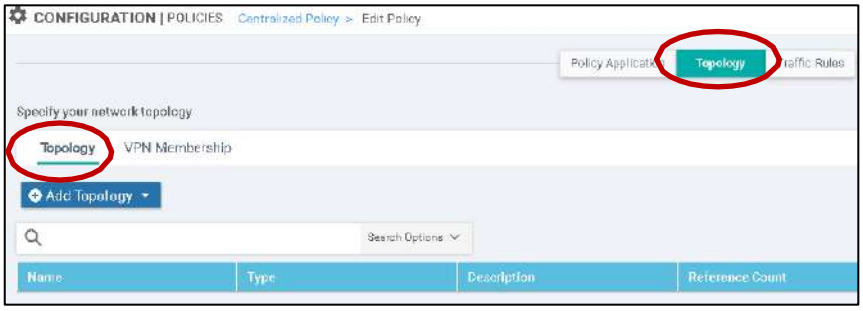
Step 26 You'll be redirected to the Topology summary window. If necessary you can edit the policy from here.

Step 27 Go to **Configuration > Policies** and click **…** to edit the Centralized policy **Central-Policy-32**.



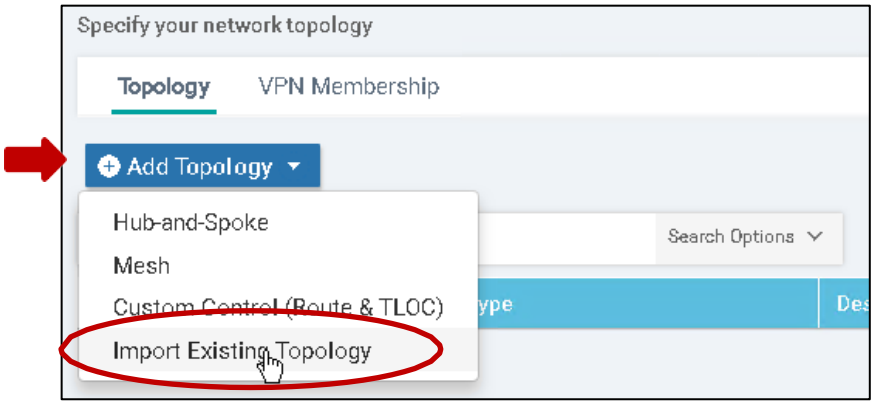
Step 28 Select **Topology** from the menu at the top of the window.

Select **Topology** from the Policies menu (default)**.**



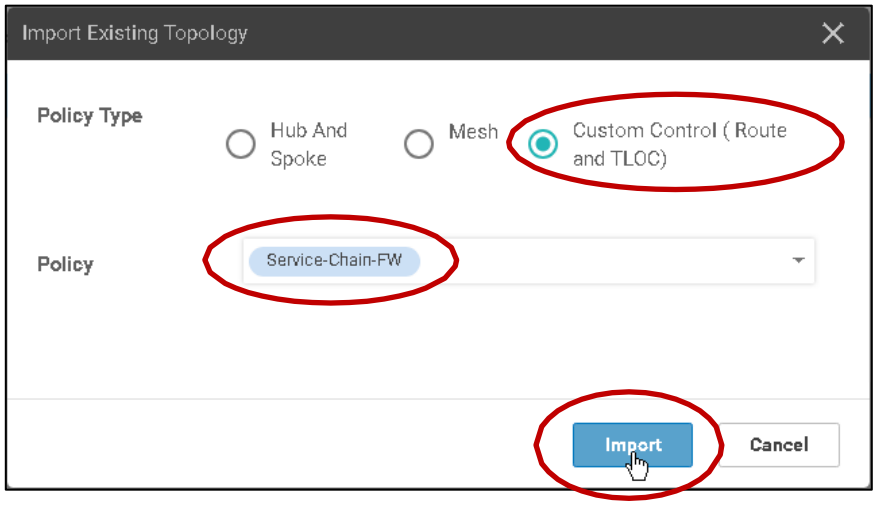
 © 2020 NterOne Corporation

Step 29 Click **+ Add Topology**, and then select **Import Existing Topology**.



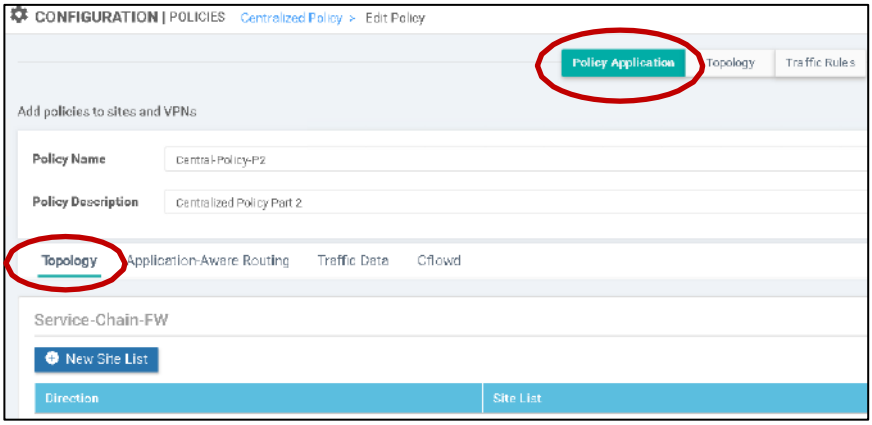
Step 30 Select **Custom Control (Route and TLOC)** for Policy Type.

Select **Service-Chain-FW** from the Policy drop-down list, and then click **Import**.



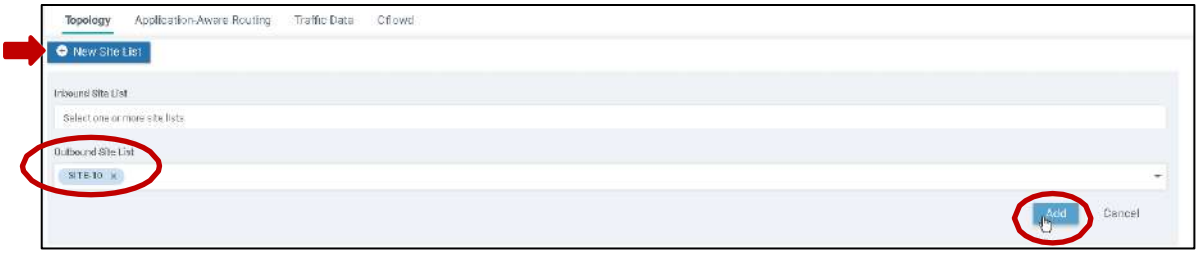
Step 31 The **Service-Chain-FW Policy will be imported** and added to **Central-Policy-32.**

Step 32 At the top of the window select **Policy Application** and on the policy menu select **Topology** (default).



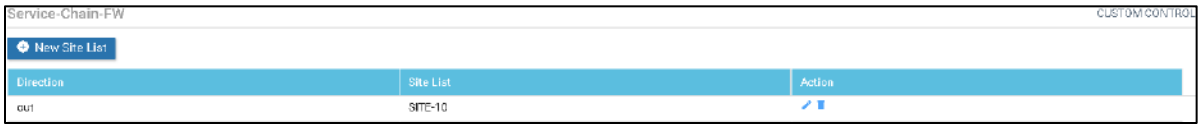
Step 33 Click + New Site List (arrow).

Select **SITE-10** from the **Outbound Site List** drop-down. Click **Add** when done.



Note The list **SITE-10** was created in the previous task.

Step 34 The **Service-Chain-FW** Policy should now be applied to **Site 10**.



Step 35 Click **Preview** and verify with the output shown below. Go back if you need to edit anything.

```
policy
   sla-class LAT-50
    latency 150
   !
  app-route-policy _VPN1_LAT-50
   vpn-list VPN1
     sequence 1
      match
```

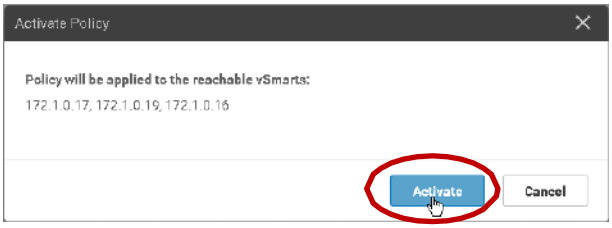
 © 2020 NterOne Corporation

```
      destination-data-prefix-list PREFIX-10-2-58-0_24
     !
     action
      sla-class LAT-50  preferred-color gold
     !
    !
    sequence 11
     match
      destination-data-prefix-list PREFIX-39-39-39-39_32
      dscp 46
     !
     action
      sla-class LAT-50  preferred-color gold
     !
    !
!
control-policy Service-Chain-FW
    sequence 1
     match route
      prefix-list PREFIX-10-2-54-0_24
     !
     action accept
      set
       service  FW vpn 1 tloc  172.1.20.2 color gold encap ipsec
      !
     !
    !
 default-action accept
!
lists
 data-prefix-list PREFIX-10-2-58-0_24
  ip-prefix 10.2.58.0/24
 !
 data-prefix-list PREFIX-39-39-39-39_32
  ip-prefix 39.39.39.39/32
 !
 prefix-list PREFIX-10-2-54-0_24
  ip-prefix 10.2.54.0/24
 !
 site-list SITE-10
```

```
    site-id 10
   !
   vpn-list VPN1
    vpn 1
   !
  !
 !
 apply-policy
  site-list SITE-10
   control-policy Service-Chain-FW out
   app-route-policy _VPN1_LAT-50
  !
 !
```

Step 36 Click **Save Policy Changes**.

Step 37 Click **Activate**.



Step 38 **Go back to branch-vE1 and verify the routing and forwarding table.**

```
branch-vE1# show ip route 10.2.54.0/24                    (Formatted for clarity)

--- LEGEND OMITTED ---

                                PROTOCOL  NEXTHOP NEXTHOP NEXTHOP
VPN  PREFIX       PROTOCOL      SUB TYPE  IF NAME ADDR    VPN TLOC IP   COLOR ENC
     STAT
-------------------------------------------------------------------------------
branch-vE1# show ip fib 10.2.54.0/24                      (Formatted for clarity)

                      NEXTHOP  NEXTHOP     NEXTHOP     SA
VPN   PREFIX          IF NAME  ADDR        LABEL       INDEX    TLOC IP        COLOR
-------------------------------------------------------------------------------
```

 © 2020 NterOne Corporation

Step 39 **It seems that something is wrong. Traffic destined to 10.2.54.0/24 appears to be forwarding to nowhere.**

Step 40 **Go to vSmart-1** and verify that **service FW is being advertised**. Use the following command.

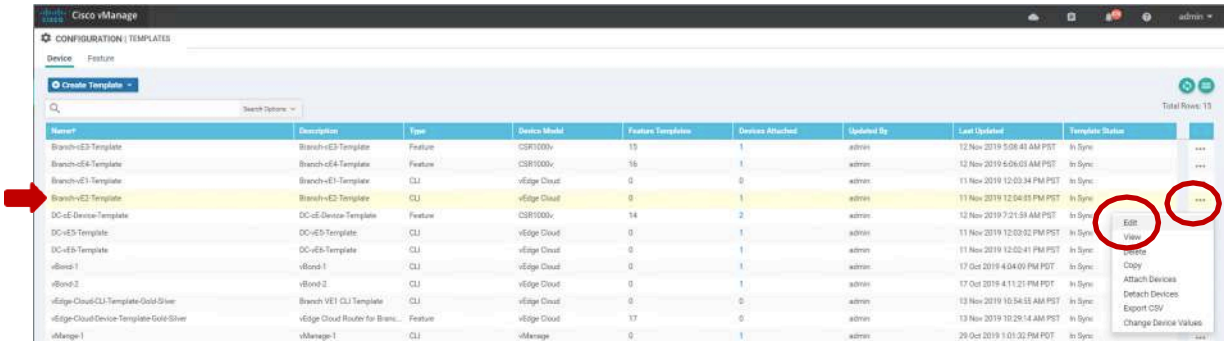
```
vSmart-1# show omp service
C  -> chosen
I  -> installed
Red -> redistributed
Rej -> rejected
L  -> looped
R  -> resolved
S  -> stale
Ext -> extranet
Inv -> invalid
Stg -> staged
U  -> TLOC unresolved


                                                PATH
VPN     SERVICE   ORIGINATOR      FROM PEER      ID      LABEL    STATUS
-----------------------------------------------------------------------
1       VPN       172.1.10.1      172.1.0.17     49      1002     C,R
                                  172.1.0.17     50      1002     C,R
                                  172.1.10.1     43      1002     C,I,R
                                  172.1.10.1     44      1002     C,I,R
1       VPN       172.1.20.2      172.1.0.17     1       1002     C,I,R
                                  172.1.0.17     11      1002     C,I,R
                                  172.1.0.19     1       1002     C,R
                                  172.1.0.19     2       1002     C,R
1       VPN       172.1.30.3      172.1.0.17     13      1002     C,R
                                  172.1.0.17     14      1002     C,R
                                  172.1.30.3     34      1002     C,I,R
                                  172.1.30.3     43      1002     C,I,R
1       VPN       172.1.30.4      172.1.0.17     5       1002     C,R
                                  172.1.0.17     16      1002     C,R
                                  172.1.30.4     34      1002     C,I,R
                                  172.1.30.4     43      1002     C,I,R
1       VPN       172.1.100.5     172.1.0.17     9       1002     C,R
                                  172.1.0.17     21      1002     C,R
                                  172.1.100.5    34      1002     C,I,R
                                  172.1.100.5    43      1002     C,I,R
```

```
1       VPN       172.1.100.6     172.1.0.19      9       1002     C,R
                                  172.1.0.19      10      1002     C,R
                                  172.1.100.6     34      1002     C,I,R
                                  172.1.100.6     43      1002     C,I,R
1       VPN       172.1.200.7     172.1.0.17      15      1002     C,I,R
                                  172.1.0.17      23      1002     C,I,R
                                  172.1.0.19      3       1002     C,R
                                  172.1.0.19      7       1002     C,R
1       VPN       172.1.200.8     172.1.0.17      25      1002     C,I,R
                                  172.1.0.17      26      1002     C,I,R
                                  172.1.0.19      13      1002     C,R
                                  172.1.0.19      14      1002     C,R
```

Step 41 There seems to be **no FW service advertised to vSmart**. This service **needs to be advertised** in order for it to be available for use.

Step 42 Go to **vManage and then to Configuration > Templates**.

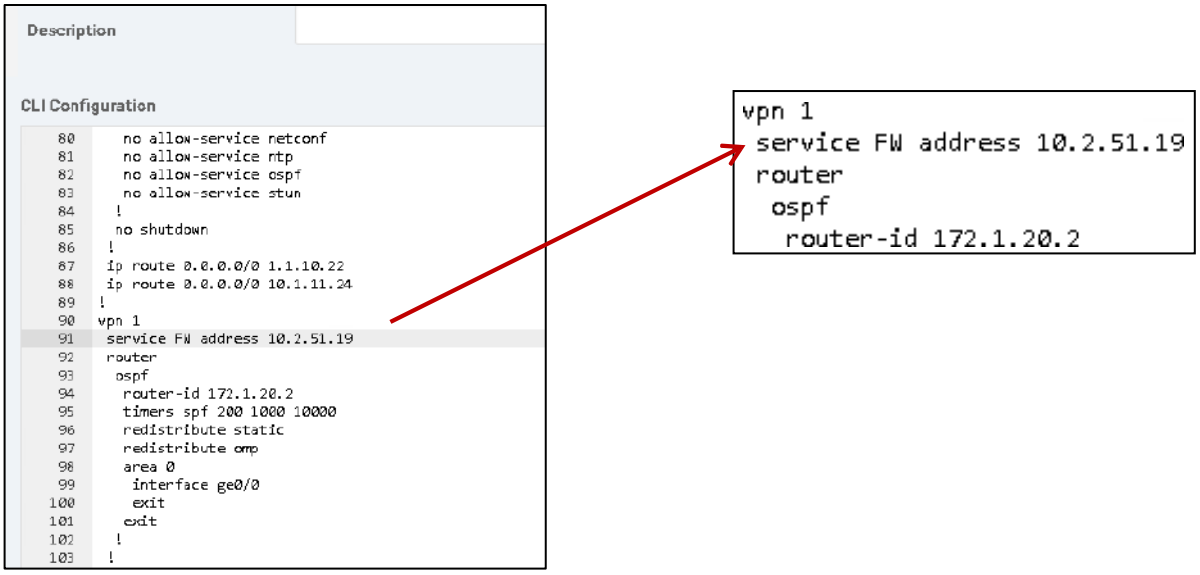
Select **Branch-vE2-Template** and then click on **…** to **edit Branch-vE2-Template (branch-vE2).** If Branch-vE2 is in CLI mode, then edit the configuration via the CLI using information in step 43.



Step 43 **On Branch-vE2** (branch-vE2) add the following line of code right below **vpn 1** in the configuration. "Service FW address 10.2.51.19"

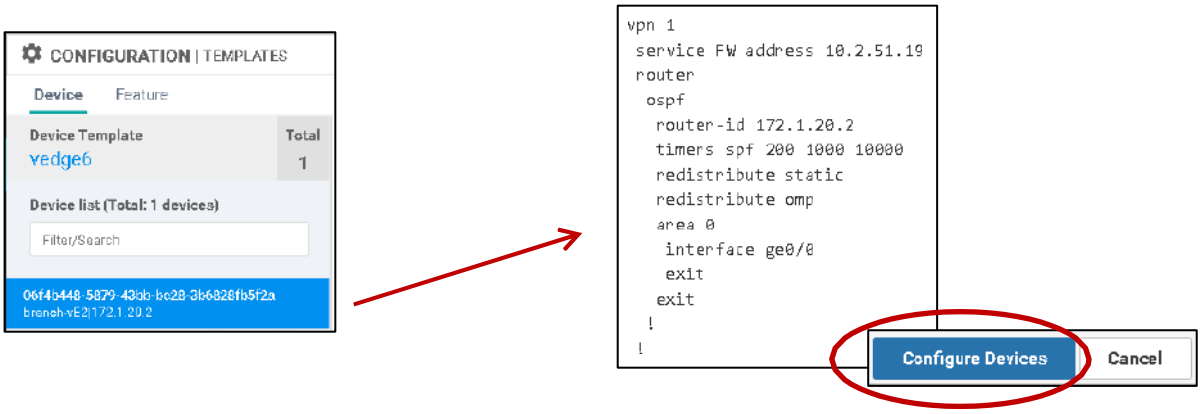
Note The Service IP address has to be reachable in order for the service to be advertised.

 © 2020 NterOne Corporation



Step 44 Click **Update**.

Step 45 Select the **device from the Device list** and confirm the configuration over to the right side. Click **Configure Devices** when done.



Step 46 **Go back to branch-vE1** and verify the results. Check the **routing and forwarding table** again.

```
branch-vE1# show ip route 10.2.54.0/24                    (Formatted for clarity)
--- LEGEND OMITTED ---
                                  PROTOCOL  NEXTHOP NEXTHOP NEXTHOP
VPN   PREFIX        PROTOCOL      SUB TYPE  IF NAME ADDR    VPN TLOC IP   COLOR ENC
      STAT
-----------------------------------------------------------------------------------
1     10.2.54.0/24 omp            -         -       -       -        172.1.20.2 gold
      ipsec F,S
```

```
branch-vE1# show ip fib 10.2.54.0/24                    (Formatted for clarity)


                      NEXTHOP    NEXTHOP      NEXTHOP    SA
VPN     PREFIX        IF NAME    ADDR         LABEL      INDEX     TLOC IP          COLOR
-----------------------------------------------------------------------------------------
1       10.2.54.0/24  ipsec      1.1.10.12    1006       187       172.1.20.2       gold
1       10.2.54.0/24  ipsec      1.1.10.12    1006       188       172.1.20.2       gold
```

Step 47 **It now appears that the Service Chain policy is in effect. Currently all traffic destined for 10.2.54.0/24** is being forwarded to the **next-hop 172.1.20.2** where an **FW service has been configured** (branch-vE2).

Step 48 **Go to branch-vE2** and confirm that the **omp service FW** is advertised to vSmart.

```
branch-vE2#          show omp services

        ----LEGEND OMITTED----

                                  PATH
        VPN     SERVICE  ORIGINATOR  FROM PEER   ID     LABEL    STATUS
        ----------------------------------------------------------------

        1       VPN       172.1.20.2 0.0.0.0     34     1002     C,Red,R
                                     0.0.0.0     43     1002     C,Red,R
        1       FW        172.1.20.2 0.0.0.0     34     1006     C,Red,R
                                     0.0.0.0     43     1006     C,Red,R
```

Verify the route table on branch-vE2.

The **next-hop TLOC IP** should be **DC-vE5,** the original destination next-hop**.**

```
branch-vE2# show ip route 10.2.54.0/24          (Formatted for clarity)

        --- LEGEND OMITTED ---
                                  PROTOCOL  NEXTHOP NEXTHOP NEXTHOP
VPN     PREFIX        PROTOCOL    SUB TYPE  IF NAME ADDR    VPN TLOC IP     COLOR  ENC   STAT
-----------------------------------------------------------------------------------------------
1       10.2.54.0/24  omp         -         -       -       -   172.1.100.5 mpls   ipsec F,S
1       10.2.54.0/24  omp         -         -       -       -   172.1.100.5 gold   ipsec F,S
```

 © 2020 NterOne Corporation

Step 49 Optionally, you can **go to one of the vSmart controllers listed** in the previous command output and confirm the FW service is advertised **from branch-vE2**.

- **e.g. 172.1.0.17** (vSmart-2)

```
vSmart-2#show omp services


--- VPN SERVICE OUTPUT OMITTED ---

1       FW        172.1.20.2   172.1.0.19       35     1006     C,R
                               172.1.0.19       36     1006     C,R
                               172.1.20.2       34     1006     C,I,R
                               172.1.20.2       43     1006     C,I,R
```

Step 50 **View the control policy applied to vSmart**. Use the following command.

```
vSmart-2# show running-config policy | begin control-policy
 control-policy Service-Chain-FW
  sequence 1
   match route
    prefix-list PREFIX-10-2-54-0_24
   !
   action accept
    set
     service FW vpn 1
     service tloc 172.1.20.2 color gold encap ipsec
    !
   !
  !
 default-action accept
!
!
```

Task 3: Create Localized QOS Policies Using Feature Templates

In this task, you will create a Localized QOS Policy using CLI.

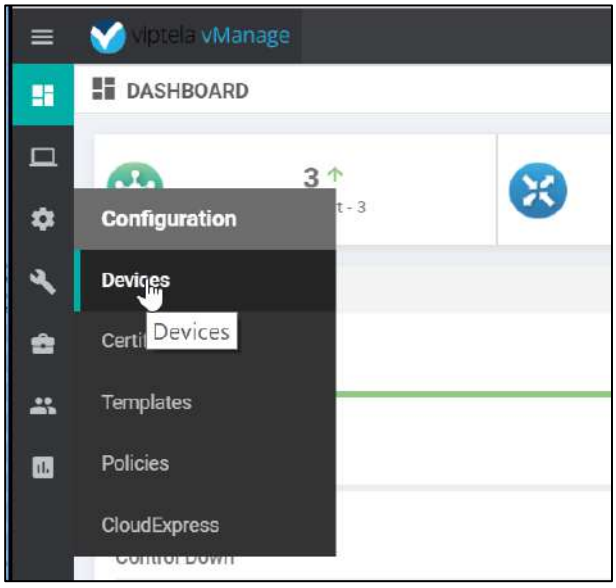
Activity Procedure

Complete these steps:

Step 1 **First, use SSH to connect to your Branch-vE1 Cloud Router.**

Step 2 **Second**, open the Chrome browser window and **login to vManage** (admin/admin).

Step 3 Go to **Configuration > Devices** and from the list find and highlight the vEdge router **branch-vE1.** Make sure to select the correct device.

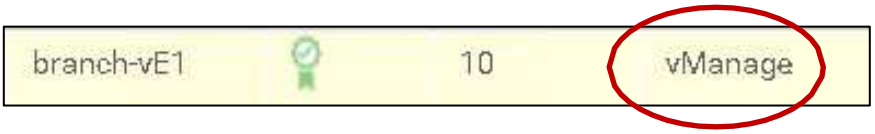


Step 4 Identify the correct vEdge router from the Device list. If the **Mode is vManage** then it needs to be changed to **CLI**.

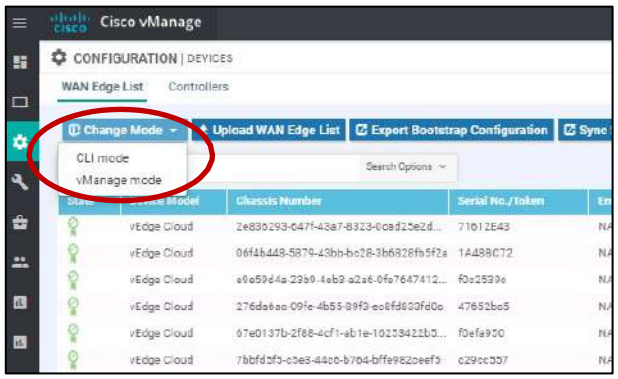
 © 2020 NterOne Corporation

Device Model	Chassis Number	Serial No./Token	Hostname	State	Site ID	Mode
vEdge Cloud	2e836293-647f-43a7-8323-0cad25e2d6...	d35cfdb3	branch-vE1		10	vManage

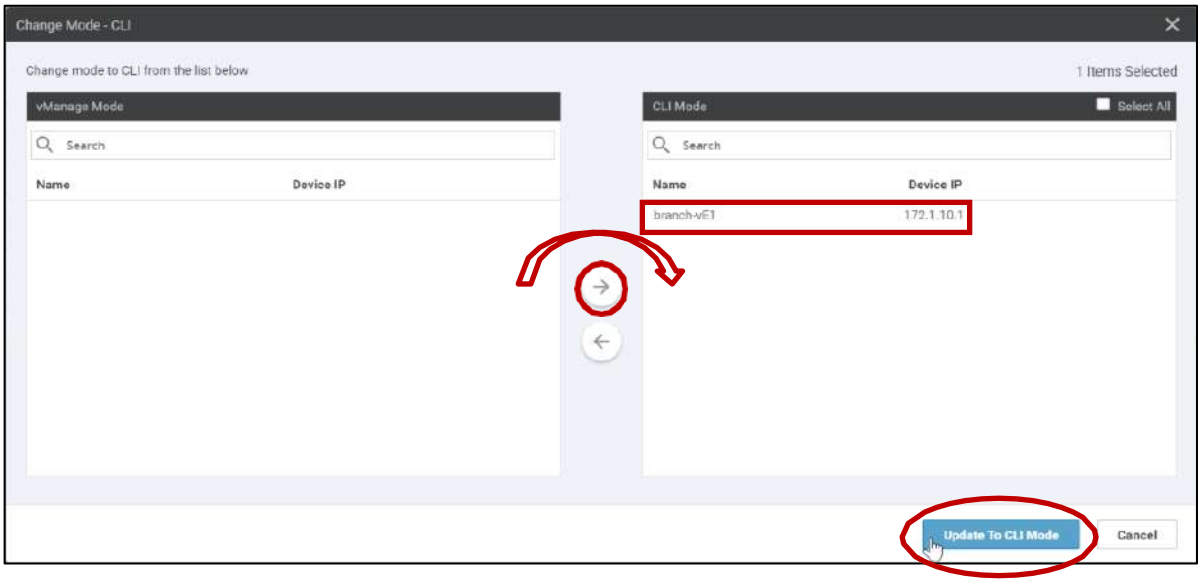
Note **When a device is in vManage mode, it means that it has a Device Template attached to it.**



Step 5 Make sure that **branch-vE1 is still selected**, then go to the top left corner of the window and click **Change Mode**, and then select **CLI mode**.



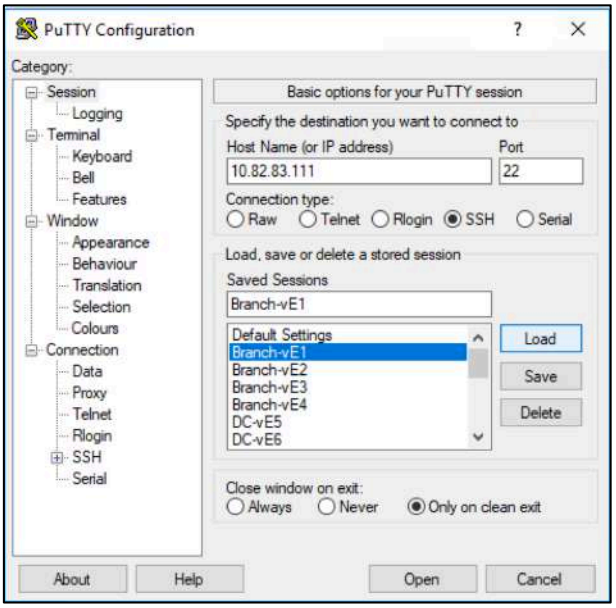
Step 6 The **Change Mode - CLI** window will appear. Select the **CORRECT** vEdge device from the **vManage Mode** list and add it to the right under **CLI Mode**. Click **Update To CLI Mode** when done.



Step 7 Wait patiently periodically pressing the refresh button

Step 8 Go back to Configuration > Device and confirm on the Devices list that it has been changed to CLI mode. Contact your instructor if you encounter any problems.

Step 9 Open Putty and find the **branch-vE1** vEdge router from the saved sessions and press the **Load** button followed by the **Open** button (Your IP address may be different than the screenshot). Press the **Yes** button to accept to open the SSH session. Login with a username of **admin** and a password of **admin.**



Step 10 First you will make a data policy that classifies incoming traffic by mapping each forwarding class to an output queue. Here, traffic classified as "be" (Best Effort) is mapped to queue 2, traffic classified as "af1" (Assured Forwarding 1) is mapped to queue 3, and so on.

Enter the following commands from the CLI:

```
branch-vE1# config
 Entering configuration mode terminal
 branch-vE1(config)# policy
 branch-vE1(config-policy)# class-map
 branch-vE1(config-class-map)# class be queue 2
 branch-vE1(config-class-map)# class af1 queue 3
 branch-vE1(config-class-map)# class af2 queue 4
 branch-vE1(config-class-map)# class af3 queue 5
 branch-vE1(config-class-map)# end
 Uncommitted changes found, commit them? [yes/no/CANCEL] yes
 Commit complete.
```

 © 2020 NterOne Corporation

```
branch-vE1# show run policy
 policy
  class-map
   class be queue 2
   class af1 queue 3
   class af2 queue 4
   class af3 queue 5
  !
 !
```

Step 11 Configure the QoS scheduler for each queue to define the importance of data packets. Depending on the priority of the traffic, you assign the bandwidth, buffer level, and random early detection (RED) drop profile associated with the queue. Here, "af3" traffic has higher priority over other traffic classes and so is configured to have 40% bandwidth and 40% buffer. Traffic in class "af2" has 30% bandwidth and 30% buffer; traffic in class "af1" class has 20% bandwidth and 20% buffer and traffic in class "be" has 10% bandwidth and 10% buffer size reflecting the respective priority of the traffic on the network. All traffic classes are configured with a drop profile of RED, meaning that instead of waiting for the queue to be full, packets are dropped randomly based on the thresholds defined.

Enter the following commands from the CLI:

```
branch-vE1(config)# policy
 branch-vE1(config-policy)# qos-scheduler af1

 branch-vE1(config-qos-scheduler-af1)# class af1
 branch-vE1(config-qos-scheduler-af1)# bandwidth-percent 20
 branch-vE1(config-qos-scheduler-af1)# buffer-percent 20
 branch-vE1(config-qos-scheduler-af1)# drops red-drop
 branch-vE1(config-qos-scheduler-af1)# exit
 branch-vE1(config-policy)# qos-scheduler af2
 branch-vE1(config-qos-scheduler-af2)# class af2
 branch-vE1(config-qos-scheduler-af2)# bandwidth-percent 30
 branch-vE1(config-qos-scheduler-af2)# buffer-percent 30
 branch-vE1(config-qos-scheduler-af2)# drops red-drop
 branch-vE1(config-qos-scheduler-af2)# exit
 branch-vE1(config-policy)# qos-scheduler af3
 branch-vE1(config-qos-scheduler-af3)# class af3
 branch-vE1(config-qos-scheduler-af3)# bandwidth-percent 40
 branch-vE1(config-qos-scheduler-af3)# buffer-percent 40
 branch-vE1(config-qos-scheduler-af3)# drops red-drop
```

```
branch-vE1(config-qos-scheduler-af3)# exit
branch-vE1(config-policy)# qos-scheduler be
branch-vE1(config-qos-scheduler-be)# class be
branch-vE1(config-qos-scheduler-be)# bandwidth-percent 10
branch-vE1(config-qos-scheduler-be)# buffer-percent 10
branch-vE1(config-qos-scheduler-be)# drops red-drop
branch-vE1(config-qos-scheduler-be)# exit
```

Step 12 Group the 4 schedulers - af1, af2, af3 and be into a QoS map called "QoS-Map" by using the following commands.

```
branch-vE1(config-policy)# qos-map QoS-Map
branch-vE1(config-qos-map-QoS-Map)# qos-scheduler af1
branch-vE1(config-qos-map-QoS-Map)# qos-scheduler af2
branch-vE1(config-qos-map-QoS-Map)# qos-scheduler af3
branch-vE1(config-qos-map-QoS-Map)# qos-scheduler be
branch-vE1(config-qos-map-QoS-Map)# exit
```

Step 13 You will next classify data packets into appropriate forwarding classes based on match conditions. Here "access-list acl1" classifies data packets originating from the host at source address 99.99.99.99 and going to the destination host at 39.39.39.39 into the "be" class. Data packets with a DSCP value of 10 in the IP header field are classified in the "af1" class, TCP packets are classified in the "af3" class, and packets going to destination port 23, which carries Telnet mail traffic, are classified in the "af2" class. All other traffic is dropped

Add the following thru the CLI:

```
branch-vE1(config-policy)# access-list ACL-1
branch-vE1(config-access-list-ACL-1)# sequence 1
branch-vE1(config-sequence-1)# match
branch-vE1(config-match)# source-ip 99.99.99.99/32
branch-vE1(config-match)# destination-ip 99.99.99.99/32
branch-vE1(config-match)# exit
branch-vE1(config-sequence-1)# action accept
branch-vE1(config-action)# class be
branch-vE1(config-action)# exit
branch-vE1(config-sequence-1)# exit
branch-vE1(config-access-list-ACL-1)# sequence 2
branch-vE1(config-sequence-2)# match
```

 © 2020 NterOne Corporation

```
branch-vE1(config-match)# dscp 10
branch-vE1(config-match)# exit
branch-vE1(config-sequence-2)# action accept
branch-vE1(config-action)# class af1
branch-vE1(config-action)# exit
branch-vE1(config-sequence-2)# exit
branch-vE1(config-access-list-ACL-1)# sequence 3
branch-vE1(config-sequence-3)# match
branch-vE1(config-match)# protocol 6
branch-vE1(config-match)# exit
branch-vE1(config-sequence-3)# action accept
branch-vE1(config-action)# class af3
branch-vE1(config-action)# exit
branch-vE1(config-sequence-3)# exit
branch-vE1(config-access-list-ACL-1)# sequence 4
branch-vE1(config-sequence-4)# match
branch-vE1(config-match)# destination-port 23
branch-vE1(config-match)# exit
branch-vE1(config-sequence-4)# action accept
branch-vE1(config-action)# class af2
branch-vE1(config-action)# exit
branch-vE1(config-sequence-4)# exit
branch-vE1(config-access-list-ACL-1)# default-action drop
branch-vE1(config-access-list-ACL-1)# end
Uncommitted changes found, commit them? [yes/no/CANCEL] yes
Commit complete.
```

Step 14 Next you will apply the access list to your interface gold vpn 1 ge0/0.

```
branch-vE1# config
Entering configuration mode terminal
branch-vE1(config)# vpn 1
branch-vE1(config-vpn-1)# interface ge0/0
branch-vE1(config-interface-ge0/0)# access-list ACL-1 out
branch-vE1(config-interface-ge0/0)# end
Uncommitted changes found, commit them? [yes/no/CANCEL] yes
Commit complete.
```

show policy qos-map-info

show policy qos-scheduler-info

Lab 10: Verifying and Troubleshooting Data Plane Issues

In this Lab, you will learn how to verify the function of the data plane of SDWAN and how to troubleshoot it.

Activity Procedure

Complete these steps:

The following devices and values will be used in this Lab.

Device Name	IP Address or URL	Username	Password
vManage-1	https://1.4.28.28 – SSH 10.82.83.131	admin	admin
branch-vE1	1.1.7.11 – SSH 10.82.83.111	admin	admin
branch-vE2	1.1.10.12 – SSH 10.82.83.112	admin	admin
vSmart-1	1.3.26.26 – SSH 10.82.83.134	admin	admin
vSmart-2	1.4.27.27 – SSH 10.82.83.135	admin	admin

 © 2020 NterOne Corporation

Task 1: Verifying the Data Plane

In this Task you will learn some useful commands and interfaces for troubleshooting the data plane

Activity Procedure

Complete these steps:

Step 1 Open putty and login to the **branch-vE1** using SSH.

Step 2 Open a second putty session to the **vSmart1** using SSH so that you have 2 open putty sessions.

Step 3 Check the control connections first to verify that they are up and running. Without the control plane, there will be no data plane. Use the **show control connections"** command. and verify output similar to the following, noting that the connections are all "**up."**

```
branch-vE1# show control connections
                                                                                PEER                                    PEER
            CONTROLLER
PEER    PEER PEER           SITE       DOMAIN PEER                              PRIV  PEER                              PUB
            GROUP
TYPE    PROT SYSTEM IP      ID         ID     PRIVATE IP                        PORT  PUBLIC IP                        PORT  LOCAL COLOR     PROXY STAT
 UPTIME     ID
-----------------------------------------------------------------------------------------------------------------------------------------------
-----------------
vsmart  dtls 172.1.0.16     1011       1      1.3.26.26                         12446 1.3.26.26                         12446 gold           No    up
  2:16:56:36  10
vsmart  dtls 172.1.0.16     1011       1      1.3.26.26                         12446 1.3.26.26                         12446 silver         No    up
  2:16:56:39  10
vsmart  dtls 172.1.0.19     1013       1      1.3.29.29                         12446 1.3.29.29                         12446 gold           No    up
  2:16:56:36  10
vsmart  dtls 172.1.0.19     1013       1      1.3.29.29                         12446 1.3.29.29                         12446 silver         No    up
  2:16:56:38  10
vmanage dtls 172.1.0.18     1001       0      1.4.28.28                         12846 1.4.28.28                         12846 silver         No    up
  0:03:08:14  0
```

Step 4 Next, you will verify if TLOC's are being received by the vEdge router from the vSmart servers by running the command "**show omp tlocs advertised | tab**". You should see output similar to the following:

```
branch-vE1# show omp tlocs advertised
 C   -> chosen
 I   -> installed
 Red -> redistributed
 Rej -> rejected
 L   -> looped
 R   -> resolved
 S   -> stale
 Ext -> extranet
 Stg -> staged
 Inv -> invalid


 ADDRESS
 FAMILY   TLOC IP          COLOR            ENCAP
 --------------------------------------------------
 ipv4     172.1.10.1       gold             ipsec
          172.1.10.1       silver           ipsec
```

```
vSmart-1# show omp tlocs advertised
 C   -> chosen
 I   -> installed
 Red -> redistributed
 Rej -> rejected

 L   -> looped
 R   -> resolved
 S   -> stale
 Ext -> extranet
 Stg -> staged
 Inv -> invalid


 ADDRESS
 FAMILY   TLOC IP          COLOR            ENCAP
 --------------------------------------------------
 ipv4     172.1.10.1       gold             ipsec
          172.1.10.1       silver           ipsec
          172.1.30.3       mpls             ipsec
          172.1.30.3       gold             ipsec
          172.1.100.6      mpls             ipsec
          172.1.100.6      gold             ipsec
          172.1.200.7      default          ipsec
          172.1.200.7      silver           ipsec
          172.1.200.8      mpls             ipsec
          172.1.200.8      silver           ipsec
```

Task 2: Verifying BFD Information

In this Task you will verify the BFD information thru the CLI and vManage GUI.

Activity Procedure

Complete these steps:

Step 1 Next, lets look at the **"show bfd sessions"** you should see information similar to the following:

 © 2020 NterOne Corporation

```
branch-vE1# show bfd sessions
                                          SOURCE TLOC      REMOTE TLOC                                  DST PUBLIC                    DST PUBLIC         DETECT
 TX
SYSTEM IP         SITE ID  STATE          COLOR            COLOR            SOURCE IP                   IP                            PORT        ENCAP  MULTIPLIER
 INTERVAL(msec) UPTIME           TRANSITIONS
-----------------------------------------------------------------------------------------------------------------------------------------------------------------
-----------------------------------------
172.1.20.2        20       up          gold             gold             1.1.7.11                    1.1.10.12                     12346       ipsec  7
 1000            2:17:06:23      0
172.1.20.2        20       up          silver           gold             1.2.9.11                    1.1.10.12                     12346       ipsec  7
 1000            2:17:06:23      0
172.1.30.3        30       up          gold             gold             1.1.7.11                    1.1.12.13                     12346       ipsec  7
 1000            2:17:05:36      0
172.1.30.3        30       up          silver           gold             1.2.9.11                    1.1.12.13                     12346       ipsec  7
 1000            2:17:05:36      0
172.1.30.4        30       up          gold             gold             1.1.7.11                    1.1.13.4                      12346       ipsec  7
 1000            2:17:05:16      0
172.1.30.4        30       up          silver           gold             1.2.9.11                    1.1.13.4                      12346       ipsec  7
 1000            2:17:05:16      0
172.1.100.5       100      up          gold             gold             1.1.7.11                    1.1.15.15                     12346       ipsec  7
 1000            2:17:06:23      0
172.1.100.5       100      up          silver           gold             1.2.9.11                    1.1.15.15                     12346       ipsec  7
 1000            2:17:06:23      0
172.1.100.6       100      up          gold             gold             1.1.7.11                    1.1.17.16                     12346       ipsec  7
 1000            2:17:06:19      0
172.1.100.6       100      up          silver           gold             1.2.9.11                    1.1.17.16                     12346       ipsec  7
 1000            2:17:06:19      0
172.1.200.7       200      up          gold             silver           1.1.7.11                    1.2.19.17                     12346       ipsec  7
 1000            2:17:05:45      0
172.1.200.7       200      up          gold             default          1.1.7.11                    10.1.20.17                    12346       ipsec  7
 1000            2:17:05:38      0
172.1.200.7       200      up          silver           silver           1.2.9.11                    1.2.19.17                     12346       ipsec  7
 1000            2:17:05:45      0
172.1.200.7       200      up          silver           default          1.2.9.11                    10.1.20.17                    12346       ipsec  7
 1000            2:17:05:38      0
172.1.200.8       200      up          gold             silver           1.1.7.11                    1.2.21.18                     12346       ipsec  7
 1000            2:17:05:27      0
172.1.200.8       200      up          silver           silver           1.2.9.11                    1.2.21.18                     12346       ipsec  7
 1000            2:17:05:27      0
```

Note Notice you can see the number of transitions that have occurred as well as the details of the sessions.

Step 2 Take a couple of minutes and also look at the output for **show bfd summary, show bfd history, and show bfd tloc-summary-list**

Step 3 Next, you will look at tunnel statsistics using the **show tunnel statistics** command. Note the tx-pkts and rx-pkts. By running this command multiple times you should see the numbers incrementing.

```
branch-vE1# show tunnel statistics

                                                                                                                                                TCP
TUNNEL    SOURCE                SOURCE  DEST                  LOCAL   REMOTE  TUNNEL                                                        MSS
PROTOCOL  IP        DEST IP     PORT    PORT    SYSTEM IP     COLOR   COLOR   MTU     tx-pkts   tx-octets   rx-pkts   rx-octets   ADJUST
-----------------------------------------------------------------------------------------------------------------------------------------------
ipsec     1.1.7.11  1.1.10.12   12346   12346   172.1.20.2    gold    gold    1441    463758    77921324    452151    75957897    1361
ipsec     1.1.7.11  1.1.12.13   12346   12346   172.1.30.3    gold    gold    1442    489770    82167560    478112    82718296    1362
ipsec     1.1.7.11  1.1.13.4    12346   12346   172.1.30.4    gold    gold    1438    489737    82173139    478265    82659522    1358
ipsec     1.1.7.11  1.1.15.15   12346   12346   172.1.100.5   gold    gold    1441    463758    77930121    452226    75992696    1361
ipsec     1.1.7.11  1.1.17.16   12346   12346   172.1.100.6   gold    gold    1441    463752    77879445    452165    75925037    1361
ipsec     1.1.7.11  1.2.19.17   12346   12346   172.1.200.7   gold    silver  1442    489815    82181525    478218    82751082    1362
ipsec     1.1.7.11  1.2.21.18   12346   12346   172.1.200.8   gold    silver  1442    489691    82156137    477933    82645179    1362
ipsec     1.1.7.11  10.1.20.17  12346   12346   172.1.200.7   gold    default 1442    489754    82160690    478222    82880392    1362
ipsec     1.2.9.11  1.1.10.12   12346   12346   172.1.20.2    silver  gold    1441    469480    78774023    469480    78772706    1361
ipsec     1.2.9.11  1.1.12.13   12346   12346   172.1.30.3    silver  gold    1442    497064    83324409    497063    86910906    1362
ipsec     1.2.9.11  1.1.13.4    12346   12346   172.1.30.4    silver  gold    1438    496959    83300074    496959    86883887    1358
ipsec     1.2.9.11  1.1.15.15   12346   12346   172.1.100.5   silver  gold    1441    469481    78774187    469481    78772877    1361
ipsec     1.2.9.11  1.1.17.16   12346   12346   172.1.100.6   silver  gold    1441    469472    78772760    469472    78771289    1361
ipsec     1.2.9.11  1.2.19.17   12346   12346   172.1.200.7   silver  silver  1442    497106    83332151    497106    86891494    1362
ipsec     1.2.9.11  1.2.21.18   12346   12346   172.1.200.8   silver  silver  1442    496969    83308911    496969    86889371    1362
ipsec     1.2.9.11  10.1.20.17  12346   12346   172.1.200.7   silver  default 1442    497087    83328226    497087    86913324    1362
```

Step 4 Many of the same outputs can be achieved thru the vManage GUI. For example, in vManage, you will go to **Monitor > Network** screen, select a device (vEdge router), then select **Realtime** in the left pane and select the Device Option **"BFD Sessions"** to see an output similar to the following:

Step 5 A graph of past BFD session statistics regarding jitter, latency and packet loss can be displayed in vManage. Go to **Monitor > Network** screen, select a device (vEdge router), select **Troubleshooting** in the left pane. Then select **Tunnel Health** and explore various tunnels. Make sure to select the **Chart Options** to view various statistics. Notice also that the time frame may be changed.

Task 3: Determining the Makeup of Network Traffic Using DPI

In this Task you will look at DPI (Deep Packet Inspection) and how to view and troubleshoot DPI.

Activity Procedure

Complete these steps:

Step 1 View the supported DPI applications by using the command **show app dpi supported-applications | tab** and view a portion of the long list of detectable applications. Note the FAMILY LONG NAME column.

 © 2020 NterOne Corporation

```
branch-vE1# show app dpi supported-applications | tab
 APPLICATION  FAMILY               APPLICATION LONG NAME    FAMILY LONG NAME      APP-ID
 ---------------------------------------------------------------------------------------
 ah           network_service      Authentication Header           Network Service       720
 dr           web                  Dr.dk                           Web                   2043
 dv           web                  DV.is                           Web                   1861
 hs           web                  Hs.fi (Helsingin Sanomat)       Web                   2097
 il           network_service      Internet Link            Network Service       637
 ip           network_service      Internet Protocol               Network Service       81
 ja           web                  Ja.is                           Web                   1897
 mk           web                  Mk.co.kr                        Web                   1213
 mq           application_service  IBM Websphere MQ                Application Service   312
 mt           web                  mt                              Web                   1214
(Output Omitted)
```

Step 2 Lets get more specific and look at the supported peer to peer family of protocols with the command

show app dpi supported-applications family peer_to_peer | tab

```
branch-vE1# show app dpi supported-applications family peer_to_peer | tab

                                                                FAMILY LONG   APP
 APPLICATION   FAMILY        APPLICATION LONG NAME              NAME          ID
 ---------------------------------------------------------------------------------
 abc           peer_to_peer  ABC Bittorrent client              Peer to Peer  1690
 adc           peer_to_peer  Advanced Direct Connect            Peer to Peer  1438
 utp           peer_to_peer  uTP (Micro Transport Protocol)     Peer to Peer  580
 ares          peer_to_peer  Ares                               Peer to Peer  272
 foxy          peer_to_peer  Foxy                               Peer to Peer  887
 jxta          peer_to_peer  Juxtapose                          Peer to Peer  1699
 luke          peer_to_peer  LUKE                               Peer to Peer  1698
 miro          peer_to_peer  Miro (getmiro.com)                 Peer to Peer  1548
 mute          peer_to_peer  Mute                               Peer to Peer  124
 slsk          peer_to_peer  SoulSeek                           Peer to Peer  184

(Output Omitted)
```

Step 3 Without traffic generators this next step is not very interesting but lets go thru the motions. Enter the command **show app dpi flows**. If we had real traffic on our network you would see something like the following:

```
vEdge# show app dpi flows

                                        Source  Dest
VPN  SRC IP          DST IP             Port    Port   PROTOCOL  APPLICATION  FAMILY             ACTIVE SINCE
-----------------------------------------------------------------------------------------------------------------------
1    10.192.42.2     74.125.20.95       20581   443    udp       unknown      Standard           2015-05-04T14:07:46+00:00
1    10.192.42.2     74.125.25.188      55742   5228   tcp       gtalk        Instant Messaging  2015-05-03T21:06:57+00:00
1    10.192.42.2     74.125.28.95       36597   443    tcp       google       Web                2015-05-04T14:12:43+00:00
1    10.192.42.2     74.125.28.95       36598   443    tcp       google       Web                2015-05-04T14:12:45+00:00
1    10.192.42.2     192.168.15.3       63665   53     udp       dns          Network Service    2015-05-04T14:14:40+00:00
1    10.192.42.2     216.58.192.14      40616   443    tcp       https        Web                2015-05-04T14:12:02+00:00
1    10.192.42.2     216.58.192.36      45889   443    tcp       https        Web                2015-05-04T14:14:40+00:00
1    10.192.42.2     216.58.192.36      45903   443    tcp       https        Web                2015-05-04T14:14:40+00:00
1    10.192.42.2     216.115.20.77      10000   10000  udp       sip          Audio/Video        2015-05-03T08:22:51+00:00
1    192.168.20.83   1.1.42.1           51586   22     tcp       ssh          Encrypted          2015-05-04T13:28:03+00:00
```

Task 4: Simulating Flows

In this Task you will use the simulate flows function to determine the next hop for a simulated traffic flow.

Activity Procedure

Complete these steps:

Step 1 In vManage NMS, select the **Monitor > Network** screen.

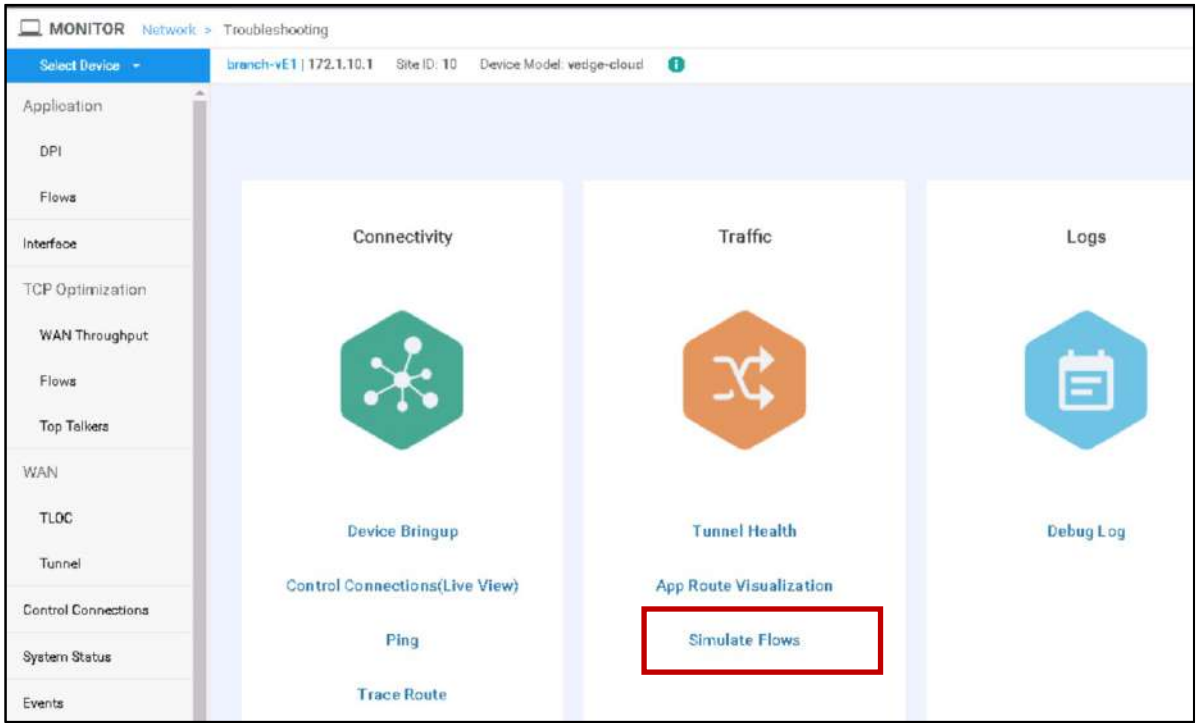
Step 2 Locate the **branch-E1** vEdge router using the Sort options drop-down and the Search box. Or scroll through the list of devices in the device table.

Step 3 Select the **branch-E1** vEdge router by clicking its hostname. The screen changes and the left pane lists the categories of information you can display about the device.

Step 4 Click **Troubleshooting** in the left pane.

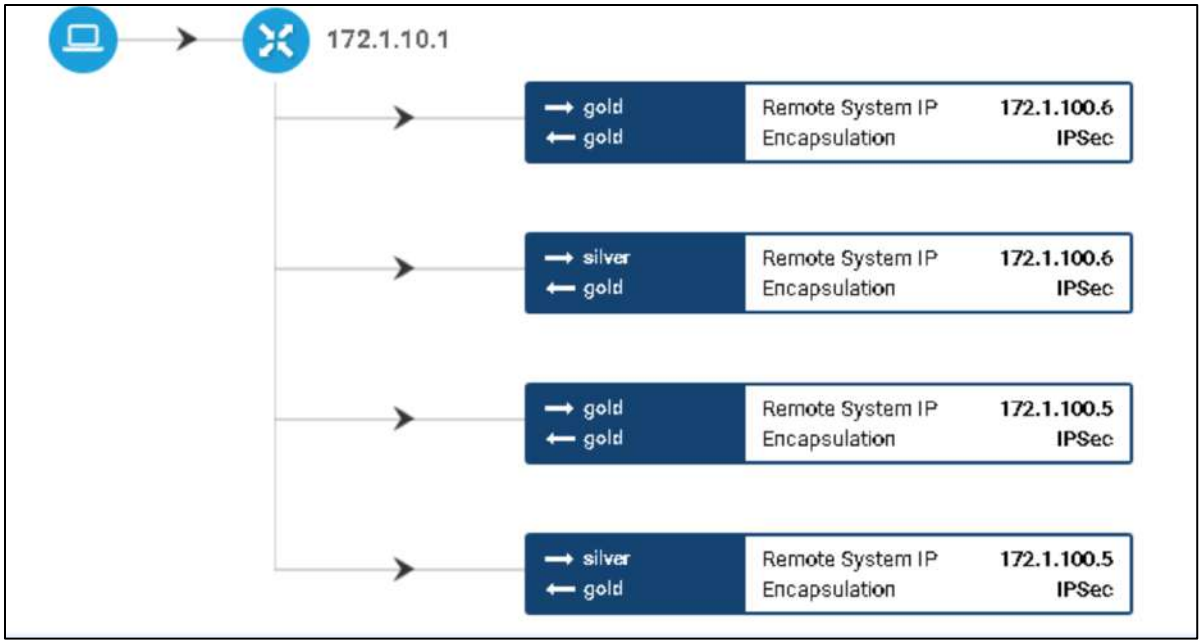
Step 5 From the Traffic pane, click **Simulate Flows**.

 © 2020 NterOne Corporation



Step 6 Set the fields to the following, open up the advanced options, then press the **Simulate** button.

- VPN* - **VPN-1**
- Source/Interface for VPN-1* - **ge0/0 – ipv4 – 10.2.50.11**
- Source IP* - **99.99.99.99**
- Destination IP – **39.3939.39**
- Advanced – Application - **Bittorrent**



Note	Notice that there are 4 different combinations that could be used for this flow

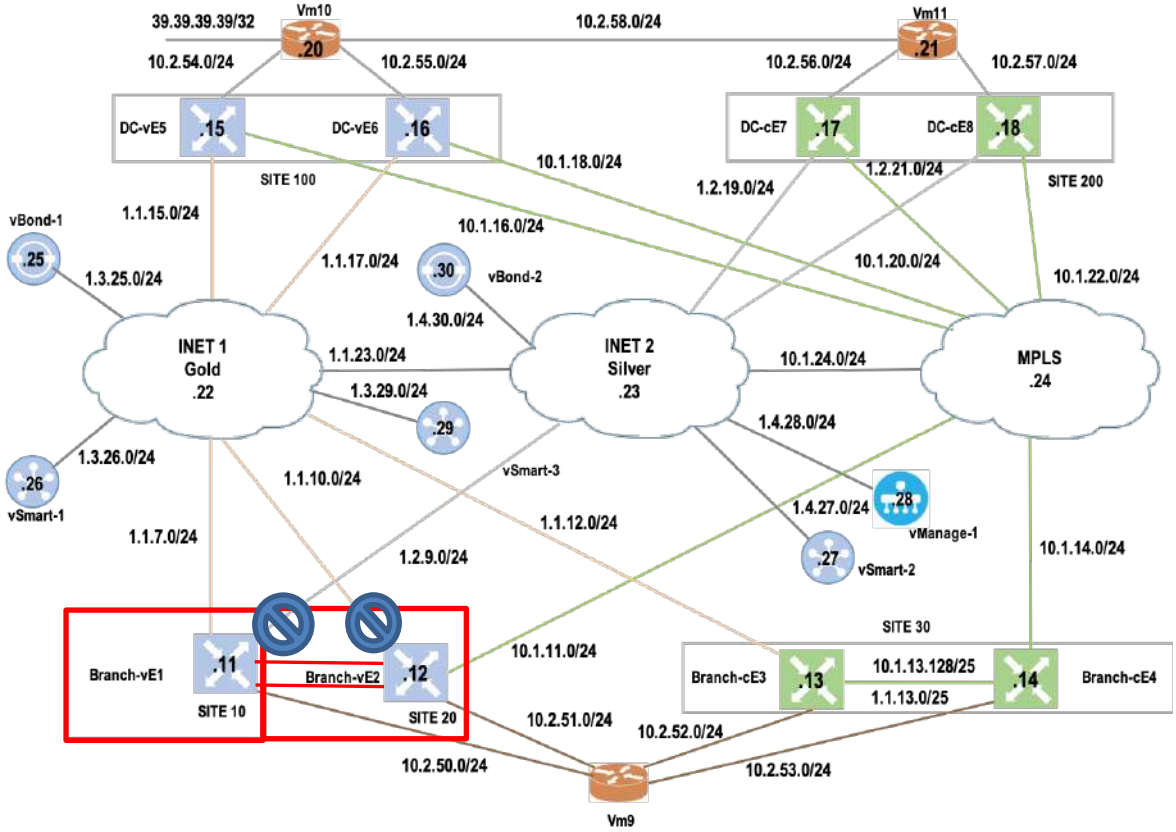
Step 7 Experiment around with other simulated flows, time permitting.

 © 2020 NterOne Corporation

Lab 11: TLOC Extension

In this Lab, you will configure layer 2 TLOC extension between Branch-vE1 and Branch-vE2. branchvE1 will be connected to the gold transport only and branch-vE2 will be connected to the MPLS transport only.

Topology



Activity Procedure

Step 1 The following devices and values will be used in this Lab.

Device Name	IP Address or URL	Username	Password
vManage-1	https://1.4.28.28 – SSH 10.82.83.131	admin	admin
Branch-vE1	1.1.7.11 – SSH 10.82.83.111	admin	admin
Branch-vE2	1.1.10.12 – SSH 10.80.88.112	admin	admin

Branch-vE1

Interface	IP Address	Default Next-Hop	Tunnel Color	VPN
ge0/1	1.1.7.11/24	0.0.0.0/0 1.1.7.22	gold	0
ge0/2	1.2.9.11/24	0.0.0.0/0 1.2.9.23	silver	0

Branch-vE2

Interface	IP Address	Default Next-Hop	Tunnel Color	VPN
ge0/1	1.1.10.12/24	0.0.0.0/0 1.1.10.22	gold	0
ge0/2	10.1.11.12/24	0.0.0.0/0 10.1.11.24	mpls	0

Perform the following steps 2 - 5 first before proceeding.

Step 2 Open an ssh session to your branch-vE1 router and login

Step 3 Enter the command **show bfd sessions | tab** and notice that you only see LOCAL TLOC Colors for the gold and silver colors.

```
branch-vE1# show bfd sessions | tab
                                 SRC    DST                    SITE  LOCAL                          DETECT      TX
SRC IP     DST IP     PROTO      PORT   PORT   SYSTEM IP       ID    COLOR   COLOR   STATE  MULTIPLIER  INTERVAL  UPTIME      TRANSITIONS
-----------------------------------------------------------------------------------------------------------------------------------
1.1.7.11   1.1.10.12  ipsec      12346  12346  172.1.20.2      20    gold    gold    up     7           1000      0:22:23:34  0
1.2.9.11   1.1.10.12  ipsec      12366  12346  172.1.20.2      20    silver  gold    up     7           1000      0:19:52:46  0
1.1.7.11   1.1.12.13  ipsec      12346  12346  172.1.30.3      30    gold    gold    up     7           1000      0:22:23:34  0
1.2.9.11   1.1.12.13  ipsec      12366  12346  172.1.30.3      30    silver  gold    up     7           1000      0:19:52:46  0
1.1.7.11   1.1.13.4   ipsec      12346  12346  172.1.30.4      30    gold    gold    up     7           1000      0:22:23:34  0
1.2.9.11   1.1.13.4   ipsec      12366  12346  172.1.30.4      30    silver  gold    up     7           1000      0:19:52:46  0
1.1.7.11   1.1.15.15  ipsec      12346  12346  172.1.100.5     100   gold    gold    up     7           1000      0:22:23:34  0
1.2.9.11   1.1.15.15  ipsec      12366  12346  172.1.100.5     100   silver  gold    up     7           1000      0:19:52:46  0
1.1.7.11   1.1.17.16  ipsec      12346  12346  172.1.100.6     100   gold    gold    up     7           1000      0:22:23:34  0
1.2.9.11   1.1.17.16  ipsec      12366  12346  172.1.100.6     100   silver  gold    up     7           1000      0:19:52:46  0
1.1.7.11   1.2.19.17  ipsec      12346  12346  172.1.200.7     200   gold    silver  up     7           1000      0:22:05:00  1
1.2.9.11   1.2.19.17  ipsec      12366  12346  172.1.200.7     200   silver  silver  up     7           1000      0:19:52:46  0
1.1.7.11   1.2.21.18  ipsec      12346  12346  172.1.200.8     200   gold    silver  up     7           1000      0:22:05:00  1
1.2.9.11   1.2.21.18  ipsec      12366  12346  172.1.200.8     200   silver  silver  up     7           1000      0:19:52:46  0
```

Step 4 Open a second ssh session to your branch-vE2 router and login

Step 5 Enter the command **show bfd sessions | tab** and notice that you only see LOCAL TLOC Color for the gold and MPLS

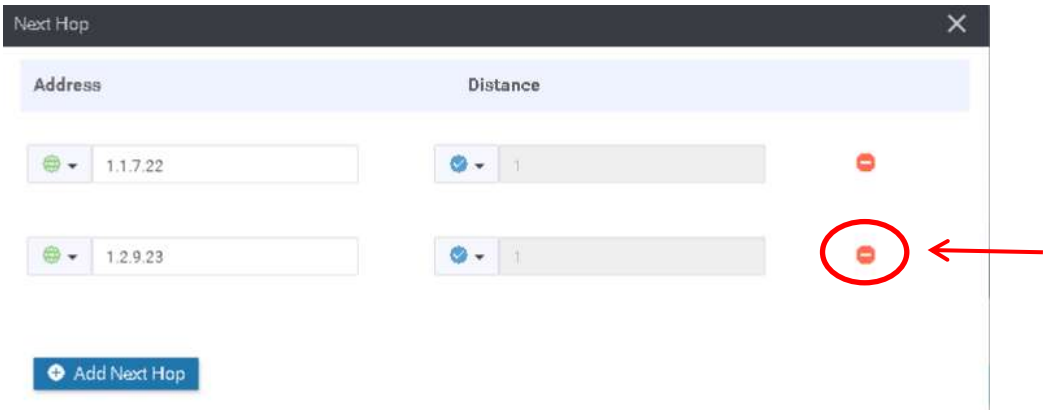
```
branch-vE2# show bfd sessions | tab
                                 SRC    DST                    SITE  LOCAL                          DETECT      TX
SRC IP     DST IP     PROTO      PORT   PORT   SYSTEM IP       ID    COLOR   COLOR   STATE  MULTIPLIER  INTERVAL  UPTIME      TRANSITIONS
-----------------------------------------------------------------------------------------------------------------------------------
1.1.10.12  1.1.12.13  ipsec      12346  12346  172.1.30.3      30    gold    gold    up     7           1000      0:22:49:40  4
10.1.11.12 10.1.13.133 ipsec     12406  12426  172.1.30.3      30    mpls    mpls    up     7           1000      0:02:40:23  0
1.1.10.12  1.1.13.4   ipsec      12346  12346  172.1.30.4      30    gold    gold    up     7           1000      0:22:49:40  4
10.1.11.12 10.1.14.14 ipsec      12406  12346  172.1.30.4      30    mpls    mpls    up     7           1000      0:03:05:46  0
1.1.10.12  1.1.15.15  ipsec      12346  12346  172.1.100.5     100   gold    gold    up     7           1000      1:05:28:31  0
10.1.11.12 10.1.16.15 ipsec      12406  12346  172.1.100.5     100   mpls    mpls    up     7           1000      0:03:05:47  0
1.1.10.12  1.1.17.16  ipsec      12346  12346  172.1.100.6     100   gold    gold    up     7           1000      1:05:28:32  0
10.1.11.12 10.1.18.16 ipsec      12406  12346  172.1.100.6     100   mpls    mpls    up     7           1000      0:03:05:47  0
1.1.10.12  1.1.7.11   ipsec      12346  12346  172.1.101.1     10    gold    gold    up     7           1000      0:22:25:00  1
1.1.10.12  1.2.9.11   ipsec      12346  12366  172.1.101.1     10    gold    silver  up     7           1000      0:19:54:12  2
1.1.10.12  1.2.19.17  ipsec      12346  12346  172.1.200.7     200   gold    silver  up     7           1000      0:22:06:26  1
10.1.11.12 10.1.20.17 ipsec      12406  12366  172.1.200.7     200   mpls    mpls    up     7           1000      0:03:05:47  0
1.1.10.12  1.2.21.18  ipsec      12346  12346  172.1.200.8     200   gold    silver  up     7           1000      0:22:06:26  1
10.1.11.12 10.1.22.18 ipsec      12406  12366  172.1.200.8     200   mpls    mpls    up     7           1000      0:03:05:46  0
```

 © 2020 NterOne Corporation

Step 6 If you prefer to configure TLOC Extensions with vManage GUI using Device Templates from Feature Templates (Branch-vE1-Template), and CLI Template (Branch-vE2-Template), continue with **Steps 7 – 41**. Otherwise, you may change Branch-vE1 and Branch-vE2 to CLI Mode and go directly to **Step 42**.

Step 7 First step is to copy the existing device template **"vEdge-Cloud-Device-Template-Gold-Silver"** and name it something like "**vEdge-Cloud-Device-Template-TLOC-Ext**", then attach that to branch-vE1 router. Then you will need to edit the template and **remove** the Silver transport from the template. See below. Then click **Save Changes**.

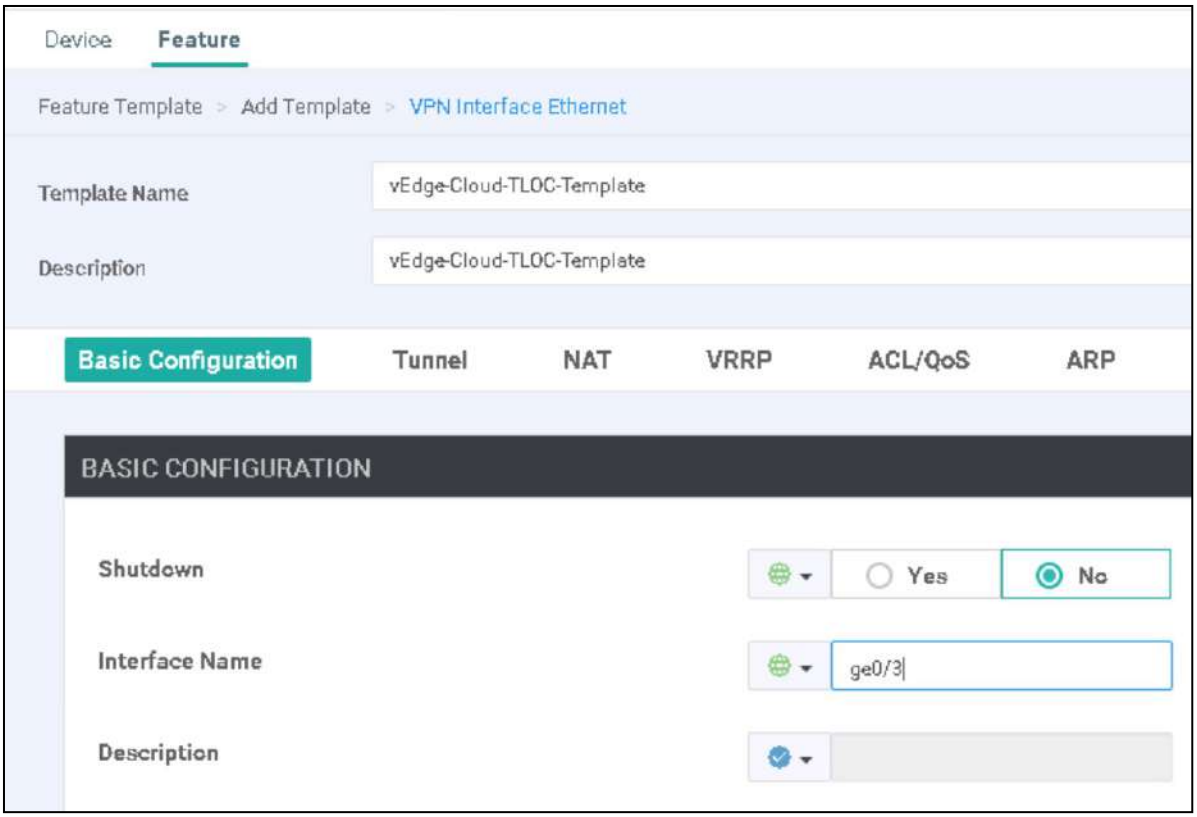
Step 8 Next, make a copy of and edit the Feature Template **"vEdge-Cloud-VPN0-Template",** name it something like **"vEdge-Cloud-VPN0-Template-TLOC-Ext"** , go down to IPV4 Route and remove the default route to the Silver transport. Then click **Save Changes**, **Save Changes**. Click **Update**, **Configure Device**, etc.



Step 9 Create a new Feature template for a vEdge Cloud router.

Step 10 Select the **VPN Interface Ethernet** feature and change the following values and **Save**:

- Template Name: vEdge-Cloud-TLOC-Template
- Description: vEdge-Cloud-TLOC-Template
- Shutdown: Change to Global and No
- Interface Name: Global and ge0/3. Click Save



Step 11 Make another new Feature template for a vEdge Cloud router.

Step 12 Select the **VPN Interface Ethernet** feature and change the following values and **Save**:

- Template Name: **vEdge-Cloud-ge0/3.10-vE1-Template**
- Description: **vEdge-Cloud-ge0/3.10-vE1-Template**
- Shutdown: Change to **Global** and **No**
- Interface Name: **Global** and **ge0/3.10**
- IPv4 Address: Change to **Device Specific**
- IP MTU: Change to **Global** and **1496**
- TLOC Extension: Change to **Global** and ge0/1

Step 13 Click **Save.**

Step 14 Make another new Feature template for a vEdge Cloud router.

Step 15 Select the **VPN Interface Ethernet** feature and change the following values and **Save**:

- Template Name: **vEdge-Cloud-ge0/3.20-vE1-Template**
- Description: **vEdge-Cloud-ge0/3.20-vE1-Template**
- Shutdown: Change to **Global** and **No**
- Interface Name: **Global** and **ge0/3.20**

 © 2020 NterOne Corporation

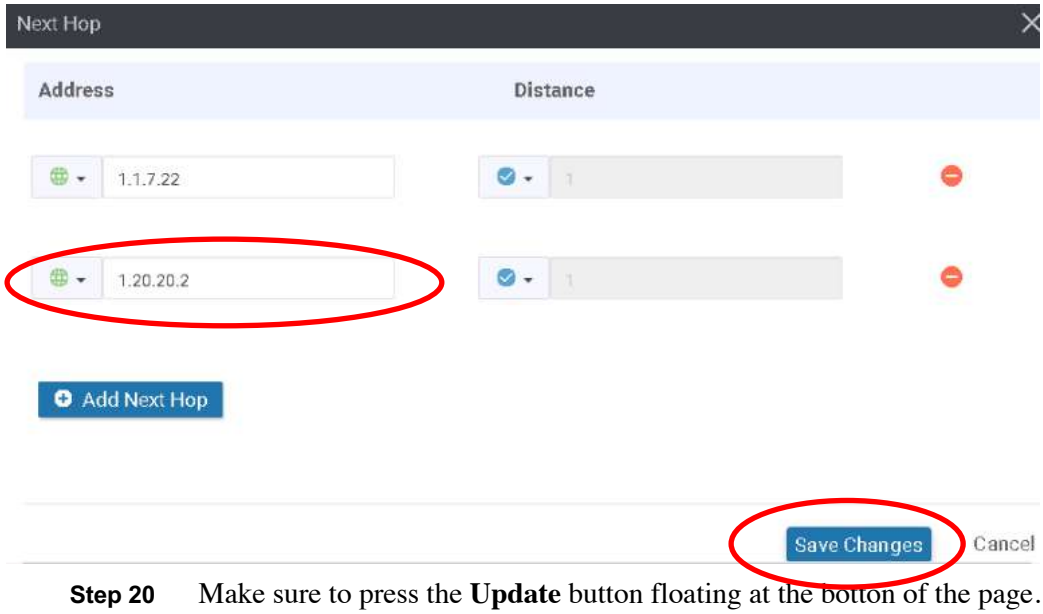
- IPv4 Address: Change to **Device Specific**
- Tunnel Interface: Change to **Global** and **On**
- Color: **mpls**
- Restrict: Change to **Global** and **On**
- IP MTU: Change to **Global** and **1496**

Step 16 Click **Save.**

Step 17 Next you need to add 1 new default route on the VPN Feature template called "**vEdge-Cloud-VPN0-Template-TLOC-Ext"**. Open up this feature template and navigate to the IPv4 section.

Step 18 Notice there is 1 Next Hop gateways, you need to add a second by pressing the pencil button on the right hand side.

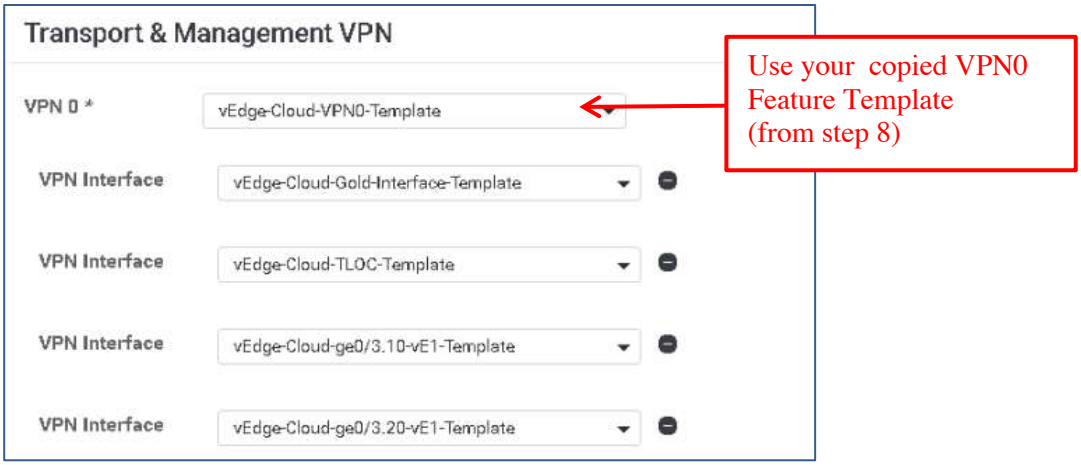
Step 19 Select the **1 Next Hop** link in blue on the pop up page and then select the **+ Add Next Hop** to add a 2nd next hop address of 1.20.20.2 then press the **Save Changes** button, then **Update**, then **Next**, then **Configure Devices**.



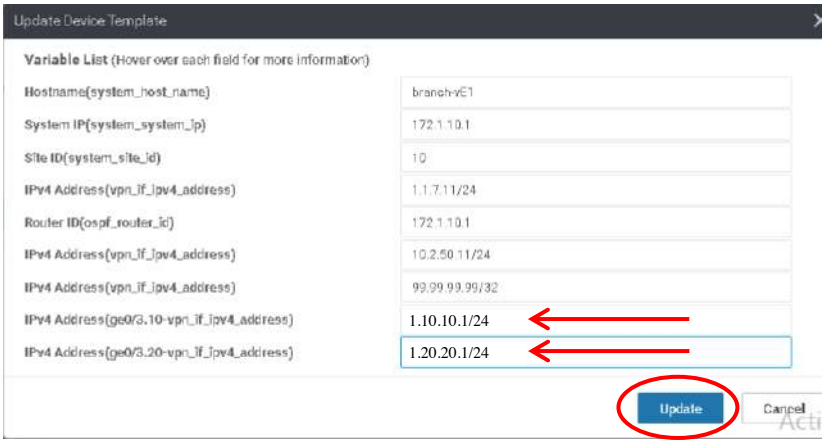
Step 20 Make sure to press the **Update** button floating at the botton of the page.

Step 21 Go the the Device template called "**vEdge-Cloud-Device-Template-TLOC-Ext**" and select the **…** and **Edit** it.

Step 22 Under the Transport and Management VPN section for VPN 0 add 3 additional VPN Interfaces by pressing the **+ VPN Interface** from the right side three times. Select the 3 feature templates configured previously and press **Update.**

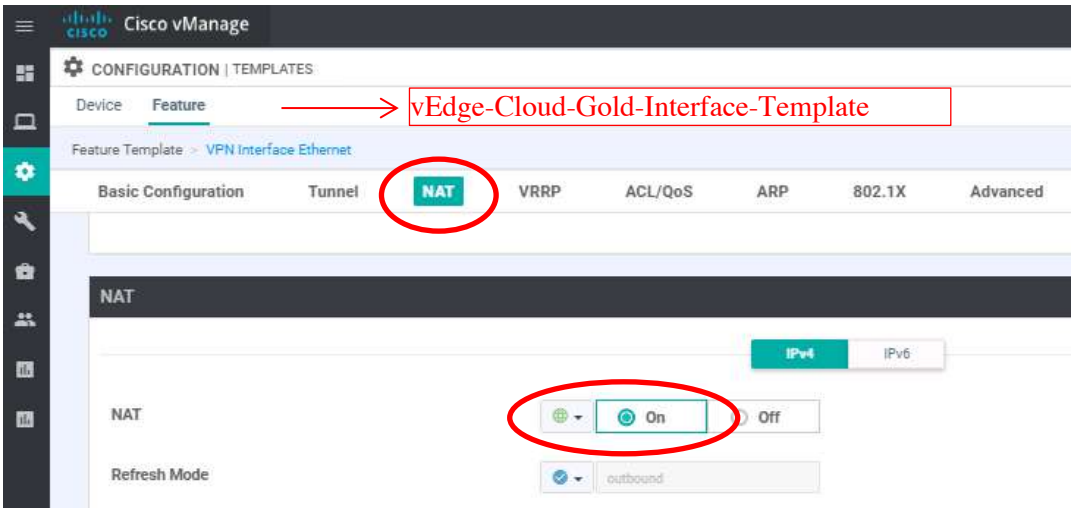


Step 23 Click **Update**, then on the Device Templates list page, click on the ellipses for your device template and attach Branch-vE1 to your device template. Subsequently, you will need to provide the **Device Specific** values for the sub-interfaces created, and click Update.



Step 24 Next step is to add the **NAT** Feature to the Device template "**vEdge-Cloud-Device-Template-TLOC-Ext**". To do this, edit the Feature Template "vEdge-Cloud-Gold-Interface-Template".

 © 2020 NterOne Corporation



Step 25 Click **Update**, **Next**, and **Configure Devices.**

Step 26 The configuration should match as follows.

```
system
  host-name                 branch-vE1
 system-ip                 172.1.10.1
 site-id                   10
 controller-group-list     10

 --- OUTPUT OMITTED ---
!
vpn 0
 dns 1.4.28.89
 interface ge0/1
  ip address 1.1.7.11/24
  nat
  tunnel-interface
   encapsulation ipsec
   color gold
  !
   --- OUTPUT OMITTED ---
  !
  no shutdown
 !
 interface ge0/3
  no shutdown
 !
 interface ge0/3.10
  ip address 1.10.10.1/24
  mtu 1496
  tloc-extension ge0/1
  no shutdown
```

```
!
interface ge0/3.20
  ip address 1.20.20.1/24
  tunnel-interface
   encapsulation ipsec
   color mpls restrict
   mtu 1496
  no shutdown
!
 ip route 0.0.0.0/0 1.1.7.22
 ip route 0.0.0.0/0 1.20.20.2
```

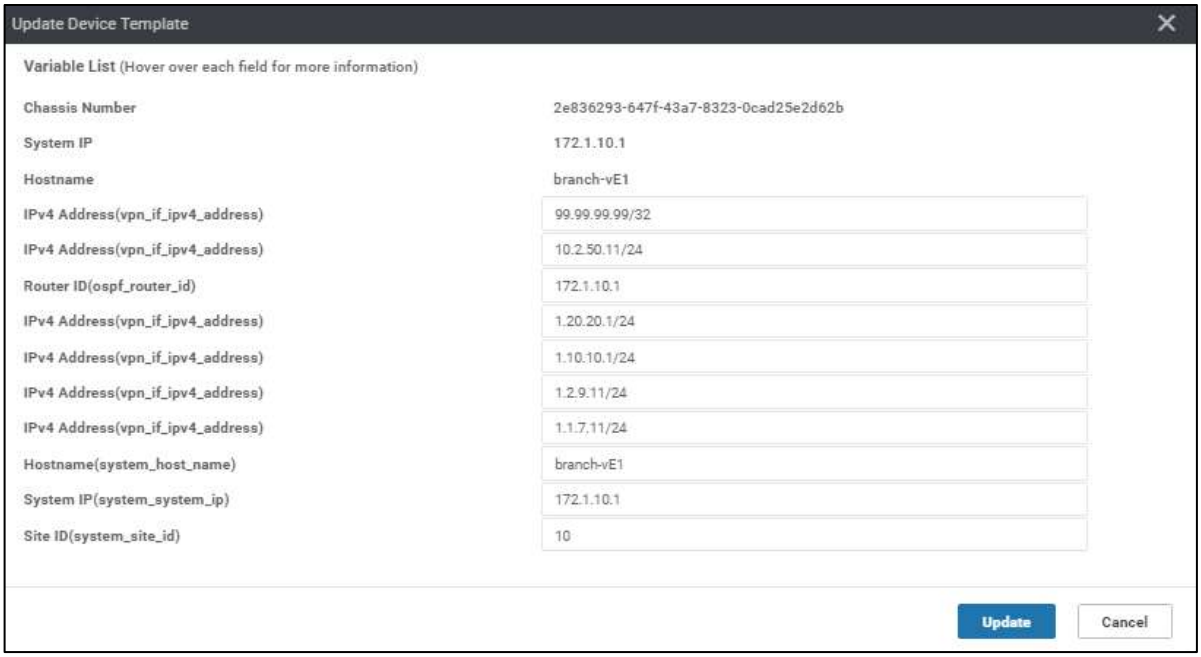
Step 27 Now lets attach the Device Template "**vEdge-Cloud-Device-Template-TLOC-Ext**" to the Branch-vE1 router (which currently has a CLI Template attached, so it must be detached first). Click on the **…** for the template and select **Attach Devices**.

Step 28 Select the **branch-vE1** router and press the **Attach** button.

Step 29 Next press the **…** to Edit Device Template and enter the following values and then press **Update**

System IP	**172.1.10.1**
Hostname(system_host_name)	**branch-vE1**
System IP(system_system_ip)	**172.1.10.1**
Site ID(system_site_id)	**10**
IPv4 Address(vpn_if_ipv4_address) ge0/1	**1.1.7.11/24**
IPv4 Address(vpn_if_ipv4_address) ge0/2	**1.2.9.11/24**
IPv4 Address(vpn_if_ipv4_address) ge0/3.10	**1.10.10.1/24**
IPv4 Address(vpn_if_ipv4_address) ge0/3.20	**1.20.20.1/24**
Router ID(ospf_router_id)	**172.1.10.1**
IPv4 Address(vpn_if_ipv4_address) ge0/0	**10.2.50.11/24**
IPv4 Address(vpn_if_ipv4_address) loopback1	**99.99.99.99/32**

 © 2020 NterOne Corporation



Step 30 Press the **Next** button and select the chassis on the left hand side.

Step 31 The configuration should match be as follows.

```
viptela-system:system
    device-model            vedge-cloud
    host-name               branch-vE1
    system-ip               172.1.10.1
    domain-id               1
    site-id                 10
    controller-group-list   10
    admin-tech-on-failure
    no route-consistency-check
    sp-organization-name    "NterOne Lab"
    organization-name       "NterOne Lab"
    console-baud-rate       115200
    vbond vbond.demo.local port 12346
    aaa
     auth-order local radius tacacs
     usergroup basic
      task system read write
      task interface read write
     !
     usergroup netadmin
     !
     usergroup operator
      task system read
      task interface read
      task policy read
      task routing read
      task security read
     !
     user admin
      password
$6$siwKBQ==$wT2lUa9BSreDPI6gB8sl4E6PAJoVXgMbgv/whJ8F1C6sWdRazdxorYYTL
rL6syiG6qnLABTnrE96HJiKF6QRq1
     !
```

```
 !
 logging
  disk
   enable
  !
 !
!
omp
 no shutdown
 graceful-restart
 advertise ospf external
 advertise connected
 advertise static
!
snmp
 no shutdown
 contact  Mike
 name     "vEdge Cloud Router"
 location "San Jose"
 view "Remote Branch"
 !
 community public
  view          "Remote Branch"
  authorization read-only
 !
!
security
 ipsec
  authentication-type sha1-hmac ah-sha1-hmac
 !
!
banner
 login "Only authorized individuals are allowed  access"
 motd  Hello
!
vpn 0
 dns 1.4.28.89 primary
 interface ge0/1
  ip address 1.1.7.11/24
  nat
  !
  tunnel-interface
   encapsulation ipsec
   color gold
   no allow-service bgp
   allow-service dhcp
   allow-service dns
   allow-service icmp
   no allow-service sshd
   no allow-service netconf
   no allow-service ntp
   no allow-service ospf
   no allow-service stun
   allow-service https
  !
  no shutdown
 !
 interface ge0/2
  !
  shutdown
 !
 interface ge0/3
  no shutdown
```

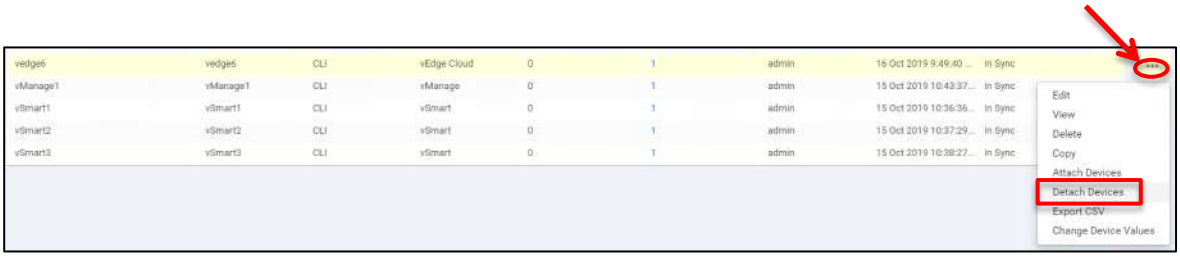
 © 2020 NterOne Corporation

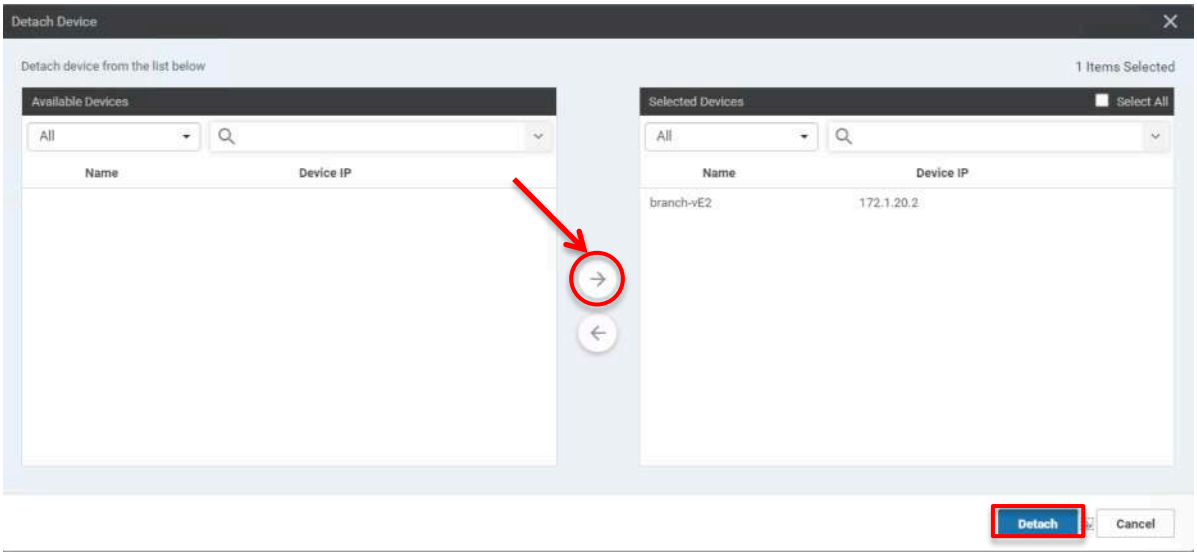
```
 !
 interface ge0/3.10
  ip address 1.10.10.1/24
  tunnel-interface
   encapsulation ipsec
   color mpls restrict
   no allow-service bgp
   allow-service dhcp
   allow-service dns
   allow-service icmp
   no allow-service sshd
   no allow-service netconf
   no allow-service ntp
   no allow-service ospf
   no allow-service stun
   allow-service https
  !
  mtu      1496
  no shutdown
 !
 interface ge0/3.20
  ip address 1.20.20.1/24
  tunnel-interface
   encapsulation ipsec
   color mpls restrict
   no allow-service bgp
   allow-service dhcp
   allow-service dns
   allow-service icmp
   no allow-service sshd
   no allow-service netconf
   no allow-service ntp
   no allow-service ospf
   no allow-service stun
   allow-service https
  !
  mtu      1496
  no shutdown
 !
 ip route 0.0.0.0/0 1.1.7.22
 ip route 0.0.0.0/0 1.20.20.2
 !
!
vpn 1
 router
  ospf
   router-id 172.1.10.1
   timers spf 200 1000 10000
   area 0
    interface ge0/0
    exit
    interface loopback1
     passive-interface
    exit
   exit
  !
 !
 interface ge0/0
  ip address 10.2.50.11/24
  no shutdown
 !
 interface loopback1
  ip address 99.99.99.99/32
```

```
    no shutdown
   !
  !
  vpn 512
   interface eth0
    ip address 10.82.83.111/20
    no shutdown
   !
  !
 !
```

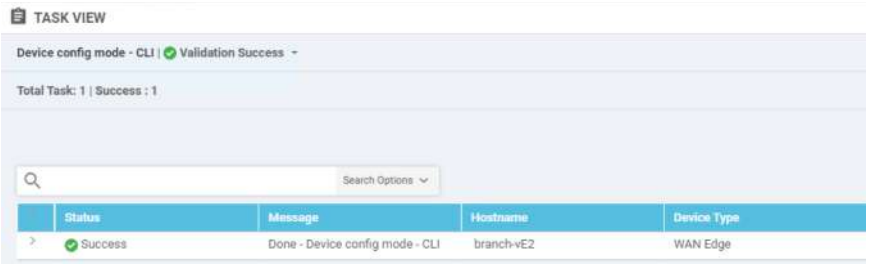
Step 32 If the configuration looks correct push the configuration to the device and ensure it is successful.

Step 33 Navigate to Conf**iguration > Templates > Device** and select the Device Template **branch-vE2** (this is a CLI template) and press the **…** to select **Detach Devices** (only if branch-vE2 is currently attached)**.** Select the **branch-vE2** device from the list of the left-hand side of the Detach Devices window and click the right-facing arrow symbol. Click the **Detach** button and wait for the Task View to complete.

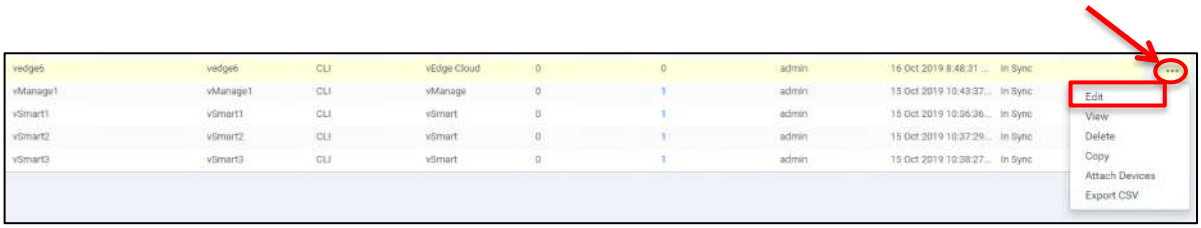




 © 2020 NterOne Corporation



Step 34 Locate the Device Template branch-vE2 and press the **…** to **Copy** it. Name the new CLI Template something like **"branch-vE2-TLOC-Extension**". Then press the … on the new template to Edit it.



Step 35 Add the following to the CLI template "vEdge2-TLOC-Extension for branch-vE2. This will need to be done manually unless you have uploaded the lab guide to the admin pc. **Leave all other parameters as they are.:**

```
vpn 0
```

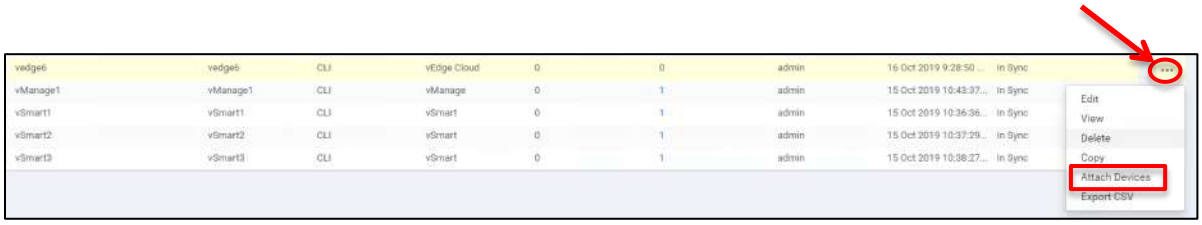
REMOVE ALL OF interface ge0/1 CONFIGURATION and shut it down (it is connected to gold transport and add "nat" to ge0/2)

```
interface ge0/1
shutdown
interface ge0/2
nat
!
interface ge0/3
no ip address
no shutdown
!
interface ge0/3.10
ip address 1.10.10.2/24
mtu 1496
tunnel-interface
encapsulation ipsec
color gold
no shutdown
```

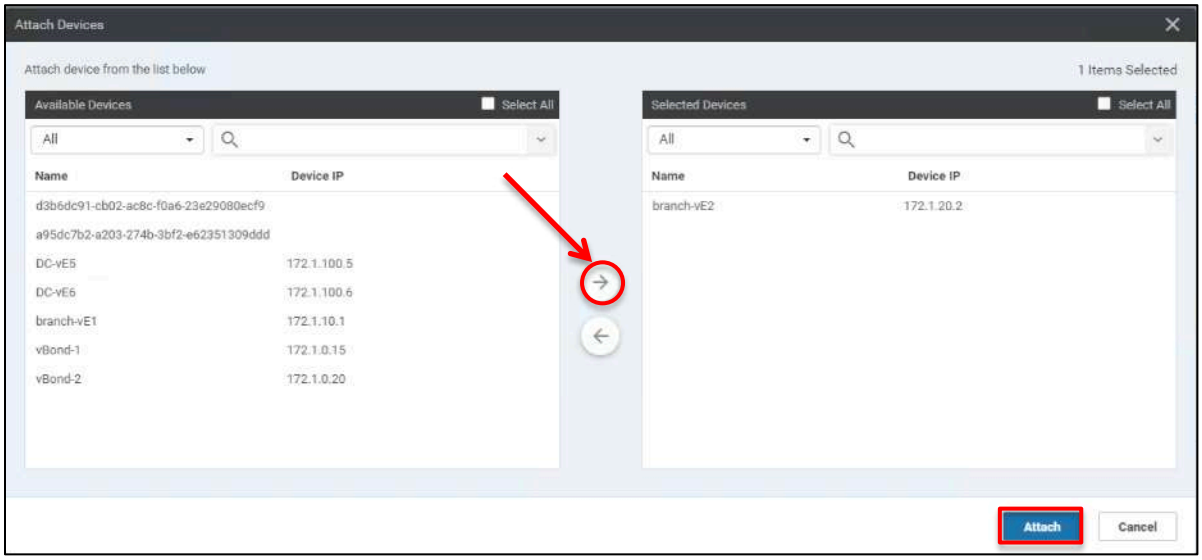
```
!
interface ge0/3.20
ip address 1.20.20.2/24
mtu 1496
tloc-extension ge0/2
no shutdown
!
no ip route 0.0.0.0/0 1.1.10.22
ip route 0.0.0.0/0 1.10.10.1
```

Step 36 Leave all other parameters alone. Click **Update**

Step 37 Navigate to Configuration > Templates > Device and select the device template "vEdge2-TLOC-Extension" and press the **…** to select **Attach Devices.**

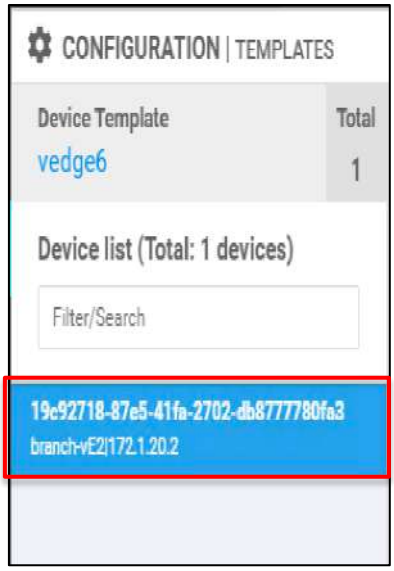


Step 38 Select the **branch-vE2** device from the list on left of the Attach Devices window, then click the right-facing arrow symbol and click the **Attach** button.



Step 39 Select the **branch-vE2** device from the Device List on left of the Device Template window, then review the CLI configuration.

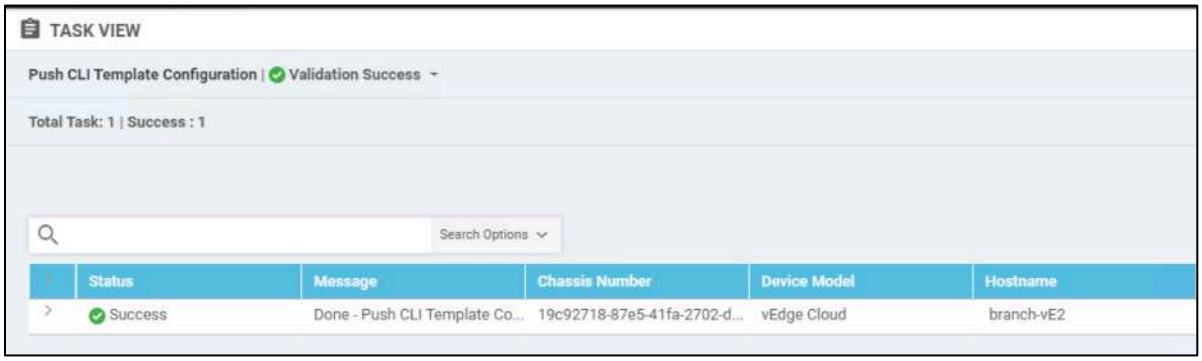
 © 2020 NterOne Corporation




```
viptela-system:system
    device-model            vedge-cloud
    host-name               branch-vE2
    system-ip               172.1.20.2
    site-id                 20
    controller-group-list   10
    admin-tech-on-failure
    route-consistency-check
    sp-organization-name    "NterOne Lab"
    organization-name       "NterOne Lab"
    upgrade-confirm         5
    vbond vbond.demo.local port 12346
    aaa
     auth-order local radius tacacs
     usergroup basic
      task system read write
      task interface read write
     !
     usergroup netadmin
     !
     usergroup operator
      task system read
      task interface read
      task policy read
      task routing read
```

```
 task security read
!
usergroup tenantadmin
!
user admin
 password
     $6$59jaBBGXOZVDmxIS$jXZyvQRdAtWHiiRy/66ZE22vA
     x5IWYnqJ0Ln0JJeyZ7gJ08QyhdcBxuDKHckgWm/SREN2Z
     4h44IuMOvcgV6oc.


(Output Omitted)
```

Step 40 Click the **Configure Devices** button and wait for the Task View to complete.



Step 41 To verify that TLOC Extensions work, edit the template for vEdge2 and shutdown interface ge0/2. Now, enter "**show control connections**" and you should have visibility via the gold transport. You can now edit the template once again, and re-enable interface ge0/2.

Step 42 If you have completed steps 1 – 41 successfully, you have completed Lab 11. Continue to Lab 12.

Optionally you can change both vEdge1 and vEdge2 to CLI mode and paste the following configuration into each of the routers.

Branch-vE1# show running-config vpn 0

vpn0

interface ge0/1

ip address 1.1.7.11/24

 © 2020 NterOne Corporation

```
nat
tunnel-interface
encapsulation ipsec
color gold
no shutdown
!
no interface ge0/2
interface ge0/2
shutdown
!
interface ge0/3
no shutdown
!
interface ge0/3.10
ip address 1.10.10.1/24
mtu 1496
tloc-extension ge0/1
no shutdown
!
interface ge0/3.20
ip address 1.20.20.1/24
tunnel-interface
encapsulation ipsec
color mpls restrict
mtu 1496
no shutdown
ip route 0.0.0.0/0 1.1.7.22
no ip route 0.0.0.0/0 1.2.9.23
ip route 0.0.0.0/0 1.20.20.2
commit
-----------------------------------------------------------------
Branch-vE2# show running-config vpn 0
vpn 0
no interface ge0/1
```

```
interface ge0/1
shutdown
!
interface ge0/2
nat
!
interface ge0/3
no shutdown
interface ge0/3.10
ip address 1.10.10.2/24
mtu 1496
tunnel-interface
encapsulation ipsec
color gold
no shutdown
!
interface ge0/3.20
ip address 1.20.20.2/24
mtu 1496
tloc-extension ge0/2
no shutdown
!
no ip route 0.0.0.0/0 1.1.10.22
ip route 0.0.0.0/0 10.1.11.24
ip route 0.0.0.0/0 1.10.10.1
commit
```

Step 43 On Branch-vE2, enter the command **show bfd sessions** and **show control connections** on both branch-vE1 and branch-vE2 to verify connectivity. BFD sessions between vEdge1 and vEdge2 will continue to indicate a status of down, but sessions to other sites should be UP.

This concludes Lab 11.

 © 2020 NterOne Corporation

Lab 12: Troubleshooting Methods and Tools

In this Lab, you will learn general troubleshooting tools and mechanisms in the SDWAN solution.

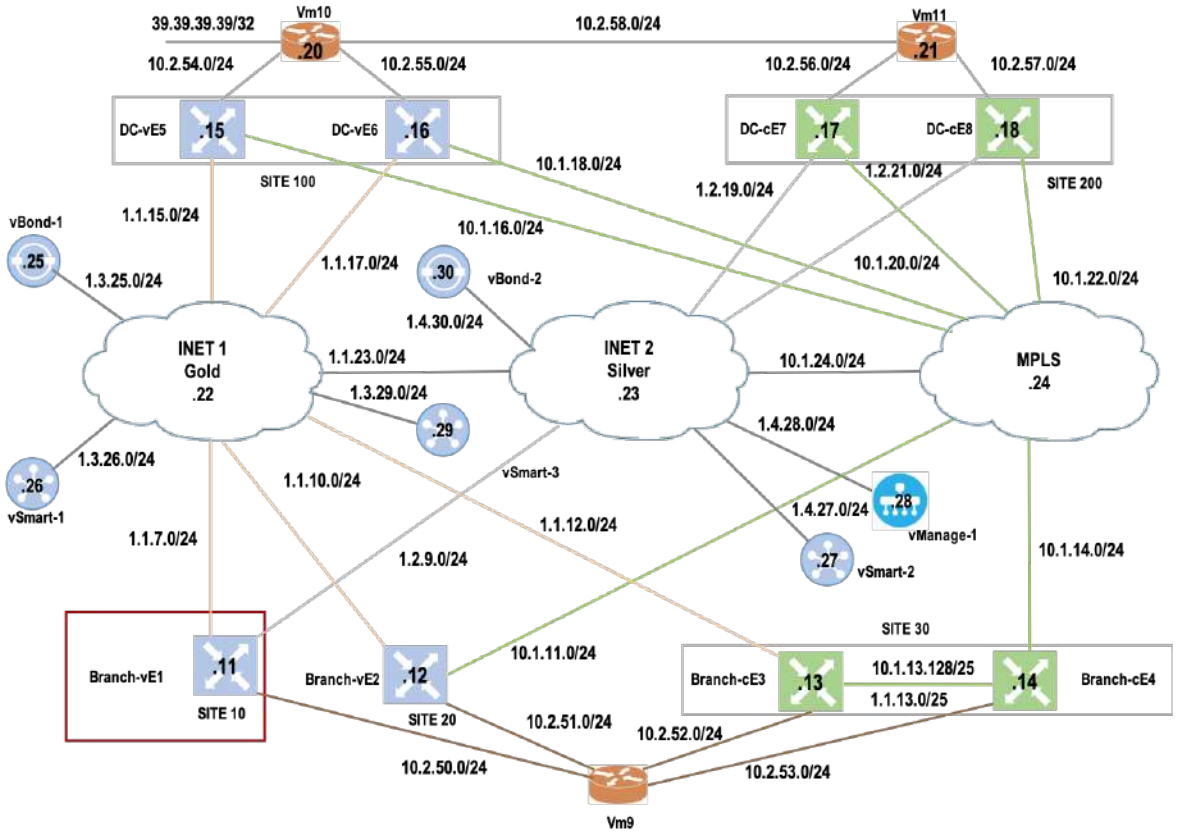
Activity Procedure

Complete these steps:

The following devices and values will be used in this Lab.

Device Name	IP Address or URL	Username	Password
vManage-1	https://1.4.28.28 – SSH 10.82.83.131	admin	admin
branch-vE1	1.1.7.11 – SSH 10.82.83.111	admin	admin
branch-vE2	1.1.10.12 – SSH 10.82.83.112	admin	admin
vSmart-1	1.3.26.26 – SSH 10.82.83.134	admin	admin
vSmart-2	1.4.27.27 – SSH 10.82.83.135	admin	admin

Topology



Task 1: Ping and Traceroute

In this Task you will deploy the ping and traceroute troubleshhoting tools within the vManage GUI.

Activity Procedure

Complete these steps:

Step 1 From the GUI of vManage you can issue a ping by navigating to **Monitor > Network** screen.

Step 2 Select the branch-vE1 device from the left-hand list by clicking its blue link.

Step 3 Click **Troubleshootin**g in the left pane.

Step 4 From the Connectivity pane, click **Ping**.

Step 5 In the Destination IP field, enter the IP address of **39.39.39.39**.

Step 6 In the VPN drop-down, select the **VPN 1** to use to reach the device.

Step 7 In the Source/Interface drop-down, select the interface **loopback1 – ipv4 - 99.99.99.99** to use to send the ping packets.

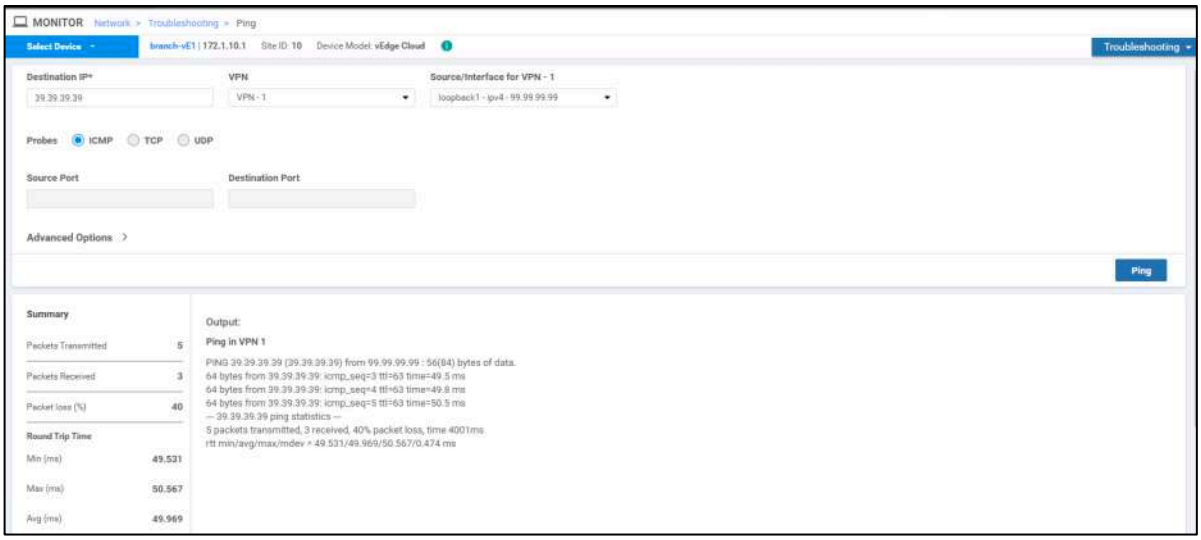
Step 8 In the Probes field, select the **ICMP** protocol type to use to send the ping packets.

 © 2020 NterOne Corporation

Step 9 Leave the Source Port, Destination Port, Type of Service and Time To Live fields blank

Step 10 Leave the Don't Fragment slider set default.

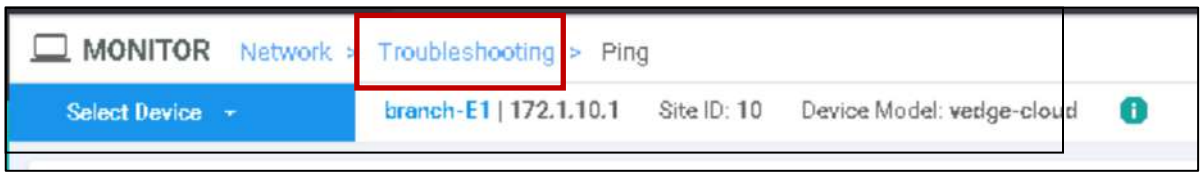
Step 11 Other advanced options like the count, payload and rapid may also be configured. Press the **Ping** button and the output should look like the response shown in the lower pane.



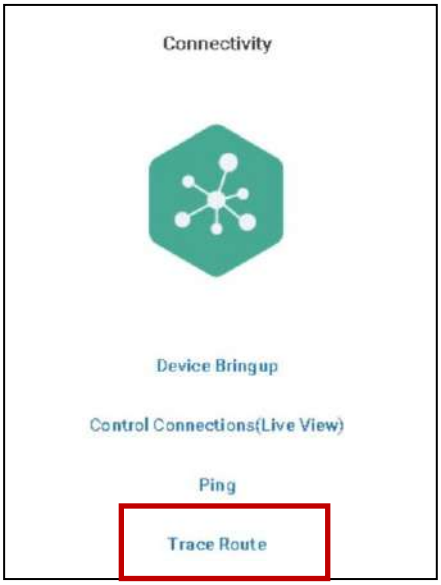
Step 12 Open a putty session to the **Branch-E1** vEdge Router, login and enter the command **ping ?**. Explore and test the CLI ping by replicating the ping you did in the vManage GUI

```
branch-vE1# ping ?
 Possible completions:
   Hostname or IP address of host to check connectivity to
   count      Number of ping packets
   fragment   do(prohibit fragmentation, even local one), dont(do not
set DF flag)
   rapid      Rapid ping
   size       Size of packets, in bytes
   source     Source interface or IP address
   vpn        VPN ID
   wait       Time to wait for a response, in seconds
```

Step 13 Back in the GUI of vManage go back up a level by selecting the Troubleshooting option.



Step 14 Select Trace Route from the Troubleshooting page.



Step 15 In the Destination IP field, enter the IP address of **39.39.39.39**.

Step 16 In the VPN drop-down, select the **VPN 1** to use to reach the device.

Step 17 In the Source/Interface drop-down, select the interface **loopback1 – ipv4 - 99.99.99.99** to use to send the traceroute packets. and then press the **Start** button.

Step 18 Go back to the putty session to the **Branch-E1** and explore the command **traceroute ?**

```
branch-vE1# traceroute ?
 Possible completions:
   help        traceroute help
   interface   Interface through which traceroute sends packets
```

 © 2020 NterOne Corporation

```
vpn          VPN ID
```

Task 2: vManage Alarms and Events

In this task you will view and verify how to use alarms and events to troubleshoot SDWAN operational and performance issues.

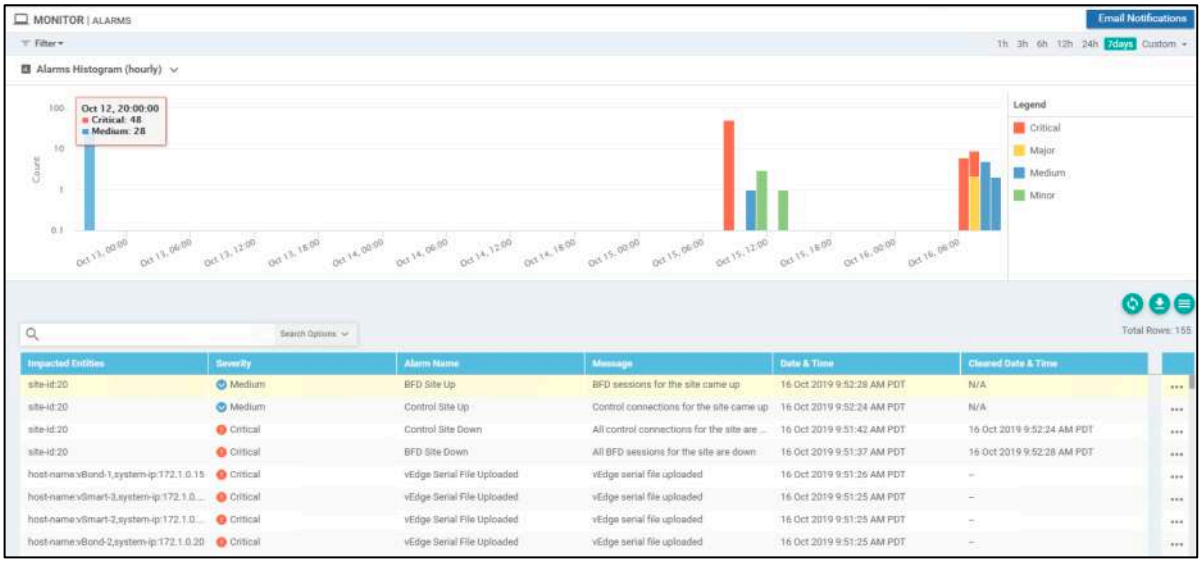
Activity Procedure

Complete these steps:

Step 1 From the vManage GUI navigate to **Monitor > Alarms**.

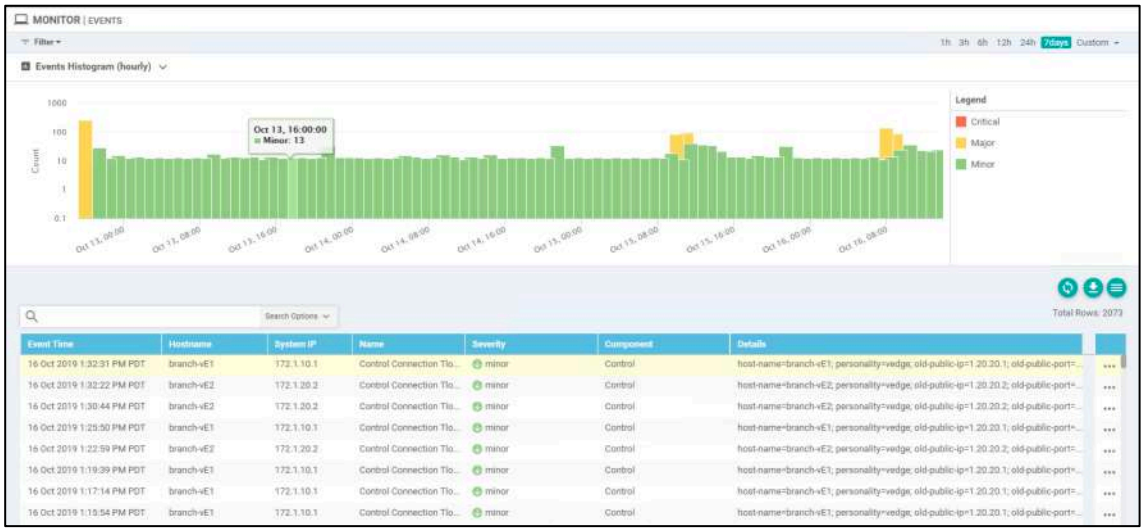
Step 2 If you do not see any alarms go to the upper right side of the interface and select **7days**, which will hopefully show some alarms. If no alarm appear you may have to simulate a failure of a wan link by issuing the following commands on the **Branch-vE1** vEdge router:

```
branch-vE1# config
Entering configuration mode terminal
branch-vE1(config)# vpn 0
branch-vE1(config-vpn-0)# interface ge0/1
branch-vE1(config-interface-ge0/1)# shut
branch-vE1(config-interface-ge0/1)# commit
Commit Complete.
branch-vE1(config-interface-ge0/1)# no shut
branch-vE1(config-interface-ge0/1)# commit
```



Step 3 You may select the "**...**" on the right side of an event to get additional details on the alarm.

Step 4 From the vManage GUI navigate to **Monitor > Events** and notice that you should have some events even if only from the interface that was shutdown and no shutdown.



Step 5 You may select the "**...**" on the right side of an event to get additional details on the event from a vSmart or vEdge device.

Step 6 Both Alarms and Events have filtering that can be performed to narrow the scope of the events displayed.

The columns may also be sorted to be ascending or descending.

Step 7 Both Alarms and Events will allow the exporting of this data to a .csv file format. Try this if it is of interest to you.

Task 3: vShell

In this Task you will investigate the vshell option within the vManage-1 CLI to view log file options.

Activity Procedure

Complete these steps:

Step 1 Login to the **vManage-1** device using SSH (10.82.83.131)

 © 2020 NterOne Corporation

Step 2 From the cli prompt enter the command **vshell**. This opens a linux shell.

Step 3 Enter the command **echo $TERM** to view the shell that is being used.

Step 4 Change the directory by using the command **cd /var/log**.

```
vManage-1# vshell
 vManage-1:~$ echo $TERM
 xterm
 vManage-1:~$ cd /var/log
 vManage-1:/var/log$
```

Step 5 List the contents of the directory by entering the command **ls**.

```
vManage-1:/var/log$ ls
auth.log            elasticsearch  messages.1  ncs        quagga                          vdebug_2018-11-04_19_57_56.tar.gz  vmware-vmsvc.log  zookeeper
auth.log.1          haproxy        messages.2  neo4j      sdavc-proxy                     vdebug_2019-09-04_14_04_55.tar.gz  vsyslog
cloud-init.log      kafka          messages.3  nginx      tallylog                        vdebug_2019-09-04_17_28_31.tar.gz  vsyslog.1
confd               kern.log       messages.4  nms        tmplog                          vdebug_2019-09-04_18_38_37.tar.gz  vsyslog.2
container-manager   lastlog        messages.5  orientdb   vconfd                          vdebug_2019-09-05_12_34_29.tar.gz  vsyslog.3
dca_log             messages       messages.6  pdb        vdebug_2018-07-13_12_27_29.tar.gz  vdebug_2019-09-06_13_47_18.tar.gz  wtmp
```

Step 6 Enter the command **exit** to leave vshell and go back to the CLI of vManage. The vShell will be used in the next task to view system log files.

Task 4: Viewing System Log Files

In this task, you will view system log files from within the vManage-1 vshell CLI.

Activity Procedure

Complete these steps:

Step 1 Login to the **vManage** device using SSH (10.82.83.131).

Step 2 From the CLI prompt, enter the command **vshell**. This opens a linux shell.

Step 3 Change the directory by using the command **cd /var/log**.

```
vManage-1# vshell
 vManage-1:~$ echo $TERM
 xterm
 vManage-1:~$ cd /var/log
 vManage-1:/var/log$
```

Step 4 List the contents of the directory by entering the command **ls**.

- **auth.log**—Login, logout, and superuser access events, and usage of authorization systems.
- **kern.log**—Kernel messages.

- **messages**—Consolidated log file that contains syslog messages from all sources.
- **vconfd**—All configuration-related syslog messages.
- **tmplog/vdebug**—All debug messages for modules whose debugging is turned on and all syslog messages above the configured priority value. Debug logging supports various levels of logging based on the module. Different modules implement the logging levels differently. For example, the system manager (sysmgr) has two logging levels (on and off), while the chassis manager (chmgr) has four different logging levels (off, low, normal, and high). You cannot send debug messages to a remote host. To enable debugging, use the **debug** operational command.
- **vsyslog**—All syslog messages from Viptela processes (daemons) above the configured priority value. The default priority value is "informational", so by default, all "notice", "warning", "error", "critical", "alert", and "emergency" syslog messages are saved.

```
vManage-1:/var/log$ ls
auth.log            elasticsearch  messages.1  ncs       quagga                          vdebug_2018-11-04_19_57_56.tar.gz  vmware-vmsvc.log  zookeeper
auth.log.1          haproxy        messages.2  neo4j     sdavc-proxy                     vdebug_2019-09-04_14_04_55.tar.gz  vsyslog
cloud-init.log      kafka          messages.3  nginx     tallylog                        vdebug_2019-09-04_17_28_31.tar.gz  vsyslog.1
confd               kern.log       messages.4  nms       tmplog                          vdebug_2019-09-04_18_38_37.tar.gz  vsyslog.2
container-manager   lastlog        messages.5  orientdb  vconfd                          vdebug_2019-09-05_12_34_29.tar.gz  vsyslog.3
dca_log             messages       messages.6  pdb       vdebug_2018-07-13_12_27_29.tar.gz  vdebug_2019-09-06_13_47_18.tar.gz  wtmp
```

Step 5 Look at the consolidated messages log file by using the command **more messages** from the shell. Take a few minutes and look at the mixture of messages present in the log file. Press the space bar for another page and CNTRL +C to exit when done.

Step 6 Next look at the authentication log file by using the command **more auth.log** from the shell. Take a little bit look at the authentication messages present in the log file. Press the space bar for another page and CNTRL +C to exit when done.

Step 7 Lets exit the shell for a minute by entering the command **exit**

Step 8 Back at the CLI for vManage let run the command **show log auth.log | tail -10** to show the 10 most recent messages in the auth.log file.

Step 9 Logout of the vManage GUI and attempt to login with a username of **BADMAN** and some random password.

Step 10 Immediately re-run the command **show log auth.log | tail -10** your output should show the failed login attempt.

```
vManage-1# show log auth.log tail 10
auth.info: Oct 16 20:30:08 vManage-1 sshd[31840]: rexec line 5: Deprecated option UsePrivilegeSeparation
auth.info: Oct 16 20:30:08 vManage-1 sshd[31840]: Accepted publickey for vmanage-admin from 172.1.0.18 port 47006 ssh2: RSA SHA256:6b+hmJHVH/YEC42z4n9Ti1L/NRO/IIJ712t
Ipx1EaFA
auth.info: Oct 16 20:30:08 vManage-1 sshd[31846]: Received disconnect from 172.1.0.18 port 47006:11: Closed due to user request.
auth.info: Oct 16 20:30:08 vManage-1 sshd[31846]: Disconnected from user vmanage-admin 172.1.0.18 port 47006
auth.info: Oct 16 20:36:39 vManage-1 sshd[13511]: rexec line 5: Deprecated option UsePrivilegeSeparation
auth.info: Oct 16 20:36:44 vManage-1 sshd[13511]: Accepted password for admin from 10.82.83.184 port 54757 ssh2
auth.info: Oct 16 20:40:08 vManage-1 sshd[20530]: rexec line 5: Deprecated option UsePrivilegeSeparation
auth.info: Oct 16 20:40:08 vManage-1 sshd[20530]: Accepted publickey for vmanage-admin from 172.1.0.18 port 47232 ssh2: RSA SHA256:6b+hmJHVH/YEC42z4n9Ti1L/NRO/IIJ712t
Ipx1EaFA
auth.info: Oct 16 20:40:08 vManage-1 sshd[20536]: Received disconnect from 172.1.0.18 port 47232:11: Closed due to user request.
auth.info: Oct 16 20:40:08 vManage-1 sshd[20536]: Disconnected from user vmanage-admin 172.1.0.18 port 47232
```

 © 2020 NterOne Corporation

Step 11 Now that you know that you can see the contents of a log file from the CLI or the Linux shell, pick your preferred way and view the **vsyslog** file (the **show log vsyslog** will show the entire log file, **tail** shows ***X*** most recent messages)

Step 12 Explore the other log files of interest; you will look at the **tmplog/vdebug** file in the next task.

Task 5: Enabling and View Debug Output

In this task, you will enable and view debug output from the branch-vE1 device CLI.

Activity Procedure

Complete these steps:

Step 1 Open a putty session to the **branch-E1** vEdge Router (10.82.83.111).

Step 2 Enter the command **debug ?** and view the many different root debugs that can be enabled.

```
branch-vE1# debug ?
 Possible completions:
   aaa            Debug AAA
   bgp            Debug BGP
   cflowd         Debug Cflowd
   chmgr          Debug chassis manager
   cloudexpress   Debug cloudexpress
   confd          Debug confd
   config-mgr     Debug configuration manager
   dbgd           Debug DBGD
   dhcp-client    Debug DHCP client
   dhcp-helper    Debug DHCP helper
   dhcp-server    Debug DHCP server
   fpm            Debug forwarding policy manager
   ftm            Debug forwarding table manager
   igmp           Debug IGMP
   iked           Debug iked
   netconf        Debug NETCONF
   omp            Debug OMP
   ospf           Debug OSPF
   pim            Debug PIM
   resolver       Debug resolver
   rtm            Debug RTM
   snmp           Debug SNMP
```

```
sysmgr          Debug system manager
tcpd            Debug tcpd
transport       Debug transport
ttm             Debug tunnel table manager
vdaemon         Debug vdaemon
vrrp            Debug VRRP
```

Step 3 Start a debug of OMP by entering the command **debug omp events level high**

Step 4 Use the command **show log tmplog/vdebug tail 100** to view the last 100 messages in the debug file.

Alternatively you can use the vShell to view the contents of the vdebug log file.

Step 5 When done with the debug make sure to turn it off by entering one of the 2 following commands:

- **no debug omp events**
- **no debug all**

Step 6 Explore other debug commands as desired; often many sub-commands exist under the root level of the commands.

 © 2020 NterOne Corporation

Lab 13: Deploy vEdge Router using ZTP (Zero Touch Provisioning) & (Troubleshooting)

In this Lab, you will deploy a vEdge Router using a ZTP Server along with troubleshooting vEdge and vBond connectivity.

Activity Procedure

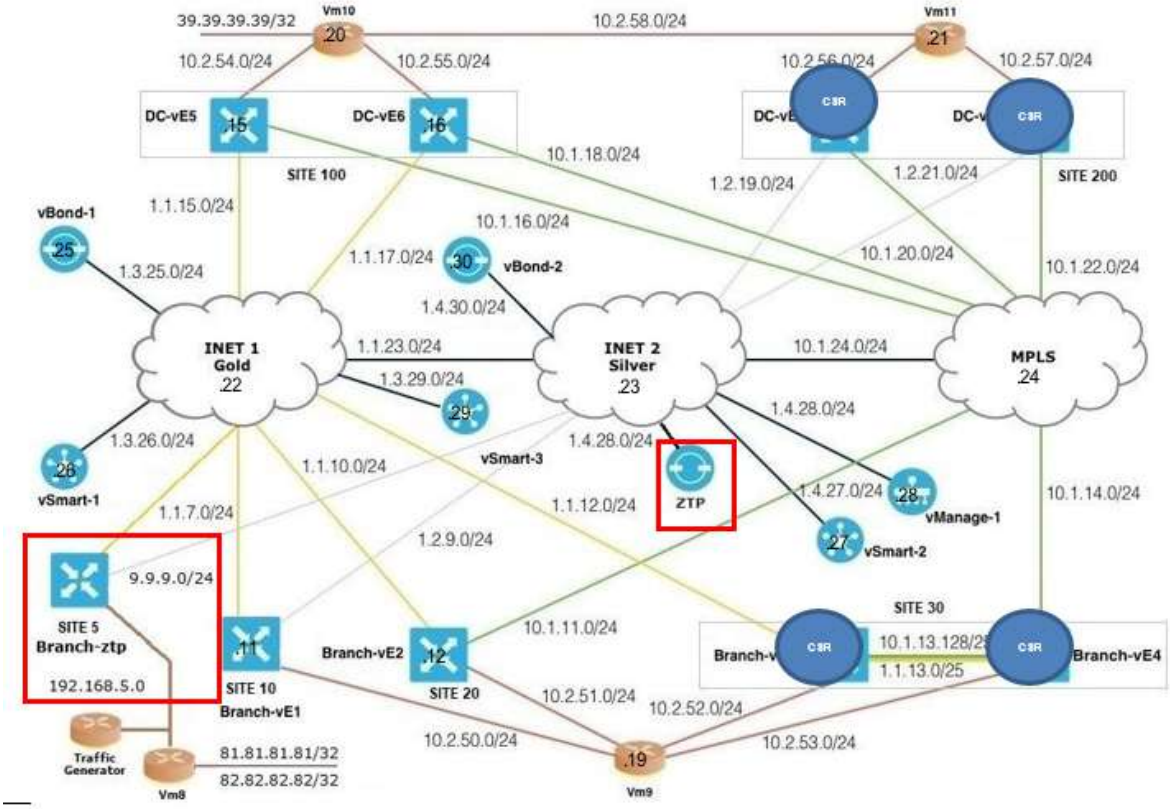
Complete these steps:

The following devices will be used in this Lab.

Device Name	IP Address or URL	Username	Password	LAB
vManage-1	https://1.4.28.28 – SSH 10.82.83.131	admin	admin	NterOne Lab
ZTP-Server	1.4.28.169 – SSH 10.82.83.169	admin	admin	NterOne Lab
Branch-ZTP	1.1.7.169 – SSH 10.82.83.115	admin	admin	NterOne Lab

Device	Interface	IP Address	Gateway	VPN	OSPF	Router-id
Branch-ZTP	ge0/0 - silver	9.9.9.169	9.9.9.23	0		
Branch-ZTP	ge0/1 – gold	1.1.7.169	1.1.7.22	0		
Branch-ZTP	ge0/2	192.168.5.169	none	1	area 0	172.1.5.1
ZTP-Server	ge0/0	1.4.28.169	1.4.28.23	0		

Topology



Task 1: Deploy ZTP Server

In this task, you will deploy a vedge-cloud router as a ZTP Server (local vBond ZTP).

Activity Procedure

Complete these steps:

Step 1 Login to the **ZTP-Server** (not Branch-ZTP) using SSH (10.82.83.169).

Step 2 Display the configuration of the ZTP Server using **"show run"**.

```
ZTP-Server# show run
system
 host-name               ZTP-Server
 admin-tech-on-failure
 no route-consistency-check
 organization-name       "NterOne Lab"
 upgrade-confirm         5
 vbond 1.4.28.169 local ztp-server
 aaa
  auth-order local radius tacacs
  usergroup basic
```

 © 2020 NterOne Corporation

```
   task system read write
   task interface read write
  !
  usergroup netadmin
  !
  usergroup operator
   task system read
   task interface read
   task policy read
   task routing read
   task security read
  !
  usergroup tenantadmin
  !
  user admin
   password
$6$VH5Y4hdKnt92/0Gm$mU9f.ne.Bhui.nblaBG7k4UFOz72n7SBpgwtX9BZz6ZUozqlwQf5NrrMwicntk
gbhMbn0Ohana0goRjSJRSc2.
  !
 !
 logging
  disk
   enable
  !
 !
!
omp
 no shutdown
 graceful-restart
 advertise connected
 advertise static
!
security
 ipsec
  authentication-type ah-sha1-hmac sha1-hmac
 !
!
vpn 0
 dns 1.4.28.89 primary
 interface ge0/0
  ip address 1.4.28.169/24
  ipv6 dhcp-client
  no shutdown
 !
 ip route 0.0.0.0/0 1.4.28.23
!
vpn 512
 interface eth0
  ip address 10.82.83.169/20
  no shutdown
 !
!
```

Step 3 If the output from your device does not match the results you see in Step 2: Enter the following **values into the CLI** or paste the contents of the **ZTP-Server.txt** file (SD-WAN Lab Files on the desktop).

```
vedge# config
Entering configuration mode terminal
vedge(config)# system
```

```
vedge(config-system)# host-name ZTP-Server
vedge(config-system)# upgrade-confirm 5
vedge(config-system)# vbond 1.4.28.169 local ztp-server
vedge(config-system)# exit
vedge(config)# vpn 0
vedge(config-vpn-0)# dns 1.4.28.89
vedge(config-vpn-0)# ip route 0.0.0.0/0 1.4.28.23
vedge(config-vpn-0)# interface ge0/0
vedge(config-interface-ge0/0)# ip address 1.4.28.169/24
vedge(config-interface-ge0/0)# no shutdown
vedge(config-interface-ge0/0)# exit
vedge(config-vpn-0)# exit
vedge(config)# show config
system
 host-name             ZTP-Server
 upgrade-confirm       5
 vbond 1.4.28.169 local ztp-server
!
vpn 0
 dns 1.4.28.89
 interface ge0/0
  ip address
  1.4.28.169/24
  no shutdown
 !
  ip route 0.0.0.0/0 1.4.28.23
 !
 vpn 512
  interface
  eth0
  ip address 10.82.83.169/20

vedge(config)# commit
ZTP-Server(config-system)# end
```

Step 4 Test connectivity to the **Gateway**.

```
ZTP-Server# ping 1.4.28.23
 Ping in VPN 0
 PING 1.4.28.23 (1.4.28.23) 56(84) bytes of data.
 64 bytes from 1.4.28.23: icmp_seq=1 ttl=64 time=12.2 ms
 64 bytes from 1.4.28.23: icmp_seq=2 ttl=64 time=20.3 ms
 64 bytes from 1.4.28.23: icmp_seq=3 ttl=64 time=19.3 ms
```

 © 2020 NterOne Corporation

```
64 bytes from 1.4.28.23: icmp_seq=4 ttl=64 time=16.3 ms
^C
--- 1.4.28.23 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3003ms
rtt min/avg/max/mdev = 12.294/17.060/20.304/3.119 ms
```

Step 5 Verify **DNS Server** connectivity and try to resolve **ztp.viptela.com.**

```
ZTP-Server# ping ztp.viptela.com
 Ping in VPN 0
 PING ztp.viptela.com (1.4.28.169) 56(84) bytes of data.
 64 bytes from ztp.viptela.com (1.4.28.169): icmp_seq=1 ttl=64
time=0.021 ms
 64 bytes from ztp.viptela.com.demo.local (1.4.28.169): icmp_seq=2
ttl=64 time=0.036 ms
 64 bytes from ztp.viptela.com (1.4.28.169): icmp_seq=3 ttl=64
time=0.037 ms
 64 bytes from ztp.viptela.com.demo.local (1.4.28.169): icmp_seq=4
ttl=64 time=0.040 ms
 64 bytes from ztp.viptela.com (1.4.28.169): icmp_seq=5 ttl=64
time=0.036 ms
 ^C
 --- ztp.viptela.com ping statistics ---
 5 packets transmitted, 5 received, 0% packet loss, time 4005ms
 rtt min/avg/max/mdev = 0.021/0.034/0.040/0.006 ms
Ping in VPN 0
PING ztp.viptela.com (1.4.28.169) 56(84) bytes of data.
64 bytes from ztp.viptela.com (1.4.28.169): icmp_seq=1 ttl=64 time=0.179
ms
```

Step 6 To ensure that we have all the correct root-ca-chain we will remove the current root-certificate and reinstall one from our vmanage NMS.

```
ZTP-Server# request root-cert-chain uninstall
Successfully uninstalled the root certificate chain
```

Step 7 This will remove the current root-cert-chain, and can be verified by using the show certificate root-ca-cert command:

```
ZTP-Server# show certificate root-ca-cert
Error: No root-ca certificate found, or no root-ca certificate
        installed
```

Step 8 Now we will install a new root-ca-chain:

```
ZTP-Server# request root-cert-chain install
scp://admin@10.82.83.131:/home/admin/ROOTCA.pem vpn 512
Uploading root-ca-cert-chain via VPN 512
Copying ... admin@10.82.83.131:/home/admin/ROOTCA.pem via VPN 512
```

```
Warning: Permanently added '10.82.83.131' (ECDSA) to the list of known hosts.

viptela 19.2.097

admin@10.82.83.131's password: admin

ROOTCA.pem
100% 1444    18.9KB/s   00:00

Installing the new root certificate chain

Successfully installed the root certificate chain

ZTP-Server#
```

Step 9 To verify that the root-ca-chain-status is installed we will execute the show control local-properties command, observe that the certificate-status is "Not-Installed":

```
ZTP-Server# show control local-properties
personality                     vedge
sp-organization-name            NterOne Lab
organization-name               NterOne Lab
root-ca-chain-status            Installed

certificate-status              Not-Installed
certificate-validity            Not Applicable
certificate-not-valid-before    Not Applicable
certificate-not-valid-after     Not Applicable

dns-name                        1.4.28.169
site-id                         0
domain-id                       1
protocol                        dtls
tls-port                        0
system-ip                       0.0.0.0
chassis-num/unique-id           a95dc7b2-a203-274b-3bf2-e62351309d11
serial-num                      No certificate installed
token                           Invalid
keygen-interval                 1:00:00:00
retry-interval                  0:00:00:16
no-activity-exp-interval        0:00:00:20
dns-cache-ttl                   0:00:02:00
port-hopped                     FALSE
time-since-last-port-hop        0:00:00:00
embargo-check                   success
number-vbond-peers              0
number-active-wan-interfaces    0
```

Step 10 Now that the root-ca-chain is installed we will proceed to install the missing certificate which is located in the /home/admin/ folder in the ztp-server vshell:

 © 2020 NterOne Corporation

```
ZTP-Server# request certificate install /home/admin/ztp-server.crt
Installing certificate via VPN 0
Copying ... /home/admin/ztp-server.crt via VPN 0
Successfully installed the certificate
ZTP-Server#
```

Notice that a **certificate has been installed**.

```
ZTP-Server# show control local-properties
personality                vedge
sp-organization-name       "NterOne Lab"
organization-name          "NterOne Lab"
certificate-status         Installed
root-ca-chain-status       Installed

<output omitted>
```

```
ZTP-Server# show certificate root-ca-cert | inc NterOne
               Issuer: C=US, ST=North Carolina, L=RTP, O=NterOne
Lab, OU=NterOne, CN=NterOne Lab/emailAddress=admin@example.com
               Subject: C=US, ST=North Carolina, L=RTP, O=NterOne
Lab, OU=NterOne, CN=NterOne Lab/emailAddress=admin@example.com
(Output Omitted)
```

Step 11 The **ZTP Server needs to be populated** with the vEdge router information that requires onboarding.

The ZTP Server acts as a **top-level vBond orchestrator**. A vEdge router will by **default try to connect to ztp.viptela.com** (pre-configured vbond value) once one of the WAN/Tunnel interfaces is assigned an IP from DHCP and becomes reachable.

Chassis Number	Serial Number	Validity
a95dc7b2-a203-274b-3bf2-e62351309ddd	2086afe3869762e2dbd773b542ba3fc1	valid
b93b51c7-c41a-9de8-fd75-bd4e2925437c	d9fcf792f32fbcc4ff5018e97ba9110d	valid
68a05544-27f6-93d6-332d-13f0fdd9bbb5	74035dc3e759fff89df85927e4f277fc	valid
a99703f6-2318-1a3d-6709-0a57ad35ed02	74035dc3e759fff89df85927e4f277fc	valid

Step 12 Using the table above **Add all of the ZTP entries** to the ZTP Server's database. Alternatively go to the **ZTP-Server.txt** file (SDWAN Lab Files folder on desktop) and paste the appropriate section from the file.

```
ZTP-Server# request device add chassis-number a95dc7b2-a203-274b-
3bf2-e62351309ddd serial-number 2086afe3869762e2dbd773b542ba3fc1
validity valid vbond vbond.demo.local org-name "NterOne Lab"
```

--- Continued ---

Step 13 **Verify the ZTP entries**. If any corrections are needed, first delete the entry with "**request device delete"** and then re-add it.

```
ZTP-Server# show ztp entries

ROOT
VBOND  ORGANIZATION  CERT
INDEX  CHASSIS NUMBER                        SERIAL NUMBER                     VALIDITY
VBOND IP          PORT   NAME          PATH
-----------------------------------------------------------------------------------------------
---------------------------------------------
2      A95DC7B2-A203-274B-3BF2-E62351309DDD  860A1C8CA6ACAB5E7323EDE94097CE3D  valid
vbond.demo.local  12346  NterOne Lab   default
3      B93B51C7-C41A-9DE8-FD75-BD4E2925437C  D9FCF792F32FBCC4FF5018E97BA9110D  valid
vbond.demo.local  12346  NterOne Lab   default
4      68A05544-27F6-93D6-332D-13F0FDD9BBB5  74035DC3E759FFF89DF85927E4F277FC  valid
vbond.demo.local  12346  NterOne Lab   default
5      A99703F6-2318-1A3D-6709-0A57AD35ED02  95A623BF38EAD7819DFF17E94CADB2E7  valid
vbond.demo.local  12346  NterOne Lab   default
```

Step 14 Use the command **"vshell"** and go to the Linux shell and verify the **Chassis Additions** in the debug log.

```
ZTP-Server# vshell

ZTP-Server:~$

ZTP-Server:~$ cd /var/log/tmplog/

ZTP-Server:/var/log/tmplog$ cat vdebug | grep "ZTP-Server vScript:"

local7.debug: Nov 14 23:46:10 ZTP-Server vScript:

local7.debug: Nov 14 23:46:10 ZTP-Server vScript: Upgrading from
16.3.0 to 18.3.0

local7.debug: Nov 14 23:46:11 ZTP-Server vScript: starting vmtoolsd
...

local7.info: Nov 14 23:46:31 ZTP-Server vScript: Starting secure-
image check timer (7 days)...

local7.debug: Nov 15 00:34:44 ZTP-Server vScript: Root ca-cert
uninstall script called:
/usr/bin/vconfd_script_uninstall_root_ca_crt_chain.sh -cli

local7.debug: Nov 15 00:34:44 ZTP-Server vScript: Successfully
uninstalled the root certificate chain

local7.debug: Nov 15 00:36:18 ZTP-Server vScript: show root cert
script called: /usr/bin/vconfd_script_show_root_ca_cert.sh -cli
```

 © 2020 NterOne Corporation

```
local7.debug: Nov 15 00:39:41 ZTP-Server vScript: Root CA certificate
chain install script called:
/usr/bin/vconfd_script_upload_root_ca_crt_chain.sh -cli path
scp://admin@10.82.83.131:/home/admin/ROOTCA.pem

local7.debug: Nov 15 00:39:41 ZTP-Server vScript: Uploading root-ca-
cert-chain via VPN 0

local7.debug: Nov 15 00:39:41 ZTP-Server vScript: Copying ...
admin@10.82.83.131:/home/admin/ROOTCA.pem via VPN 0

local7.debug: Nov 15 00:39:54 ZTP-Server vScript: Failed to install
the root certificate chain !!

local7.debug: Nov 15 00:40:03 ZTP-Server vScript: Root CA certificate
chain install script called:
/usr/bin/vconfd_script_upload_root_ca_crt_chain.sh -cli path
scp://admin@10.82.83.131:/home/admin/ROOTCA.pem vpn 512

local7.debug: Nov 15 00:40:03 ZTP-Server vScript: Uploading root-ca-
cert-chain via VPN 512

local7.debug: Nov 15 00:40:03 ZTP-Server vScript: Copying ...
admin@10.82.83.131:/home/admin/ROOTCA.pem via VPN 512

local7.debug: Nov 15 00:40:10 ZTP-Server vScript: Failed to install
the root certificate chain !!

local7.debug: Nov 15 00:40:13 ZTP-Server vScript: Root CA certificate
chain install script called:
/usr/bin/vconfd_script_upload_root_ca_crt_chain.sh -cli path
scp://admin@10.82.83.131:/home/admin/ROOTCA.pem vpn 512

local7.debug: Nov 15 00:40:13 ZTP-Server vScript: Uploading root-ca-
cert-chain via VPN 512

local7.debug: Nov 15 00:40:13 ZTP-Server vScript: Copying ...
admin@10.82.83.131:/home/admin/ROOTCA.pem via VPN 512

local7.debug: Nov 15 00:40:15 ZTP-Server vScript: Installing the new
root certificate chain

local7.debug: Nov 15 00:40:15 ZTP-Server vScript: Successfully
installed the root certificate chain

local7.debug: Nov 15 00:49:13 ZTP-Server vScript: Certificate install
script called: /usr/bin/vconfd_script_cert.sh -cli path
/home/admin/ztp-server.crt

local7.debug: Nov 15 00:49:13 ZTP-Server vScript: Installing
certificate via VPN 0

local7.debug: Nov 15 00:49:13 ZTP-Server vScript: Copying ...
/home/admin/ztp-server.crt via VPN 0

local7.debug: Nov 15 00:49:14 ZTP-Server vScript: Successfully
installed the certificate

local7.debug: Nov 15 00:51:23 ZTP-Server vScript: show root cert
script called: /usr/bin/vconfd_script_show_root_ca_cert.sh -cli

local7.debug: Nov 15 00:52:06 ZTP-Server vScript: show root cert
script called: /usr/bin/vconfd_script_show_root_ca_cert.sh -cli
```

```
local7.debug: Nov 15 00:52:10 ZTP-Server vScript: show root cert
script called: /usr/bin/vconfd_script_show_root_ca_cert.sh -cli

local7.debug: Nov 15 00:59:01 ZTP-Server vScript: vSmart chassis-
numbers add to ztp DB:
/usr/bin/vconfd_script_add_chassis_numbers_to_ztp.sh -cli chassis
a95dc7b2-a203-274b-3bf2-e62351309111 serial
2086afe3869762e2dbd773b542ba3fc1 validity valid vbond
vbond.demo.local org NterOne Lab

local7.debug: Nov 15 00:59:01 ZTP-Server vScript: Chassis number
A95DC7B2-A203-274B-3BF2-E62351309111 successfully added to the
database

local7.debug: Nov 15 01:00:43 ZTP-Server vScript: vSmart chassis-
numbers add to ztp DB:
/usr/bin/vconfd_script_add_chassis_numbers_to_ztp.sh -cli chassis
b93b51c7-c41a-9de8-fd75-bd4e2925437c serial
d9fcf792f32fbcc4ff5018e97ba9110d validity valid vbond
vbond.demo.local org NterOne Lab

local7.debug: Nov 15 01:00:43 ZTP-Server vScript: Chassis number
B93B51C7-C41A-9DE8-FD75-BD4E2925437C successfully added to the
database

local7.debug: Nov 15 01:00:51 ZTP-Server vScript: vSmart chassis-
numbers add to ztp DB:
/usr/bin/vconfd_script_add_chassis_numbers_to_ztp.sh -cli chassis
68a05544-27f6-93d6-332d-13f0fdd9bbb5 serial
74035dc3e759fff89df85927e4f277fc validity valid vbond
vbond.demo.local org NterOne Lab

local7.debug: Nov 15 01:00:51 ZTP-Server vScript: Chassis number
68A05544-27F6-93D6-332D-13F0FDD9BBB5 successfully added to the
database

local7.debug: Nov 15 01:01:00 ZTP-Server vScript: vSmart chassis-
numbers add to ztp DB:
/usr/bin/vconfd_script_add_chassis_numbers_to_ztp.sh -cli chassis
a99703f6-2318-1a3d-6709-0a57ad35ed02 serial
95a623bf38ead7819dff17e94cadb2e7 validity valid vbond
vbond.demo.local org NterOne Lab

local7.debug: Nov 15 01:01:00 ZTP-Server vScript: Chassis number
A99703F6-2318-1A3D-6709-0A57AD35ED02 successfully added to the
database

ZTP-Server:/var/log/tmplog$
```

Step 15 vshell will be used in greater detail in the next Task.

Step 16 End of Task 1.

 © 2020 NterOne Corporation

Task 2: Deploy vEdge Router Using ZTP

In this Task you will deploy a vedge-cloud router using Zero Touch Provisioning (ZTP).

Activity Procedure

Complete these steps:

Step 1 Login to **Branch-ZTP** using SSH (10.82.83.115) (not ZTP-Server).

Step 2 **Use putty on the Admin PC.**

Note **This task will require multiple putty sessions on the Admin PC to monitor ZTP transactions.**

Step 3 Display the configuration of the Branch-ZTP vEdge router using **"show run"**. The majority of the configuration is default values and will be **modified using a Template** assigned to the device from vManage.

```
vedge# show run
system
 host-name                       vedge
 admin-tech-on-failure
 no route-consistency-check
 vbond ztp.viptela.com

--- OUPUT OMITTED ---

vpn 0
 interface ge0/0
  ip dhcp-client
  ipv6 dhcp-client
  tunnel-interface
   encapsulation ipsec
   no allow-service bgp
   allow-service dhcp
   allow-service dns
   allow-service icmp
   no allow-service sshd
   no allow-service netconf
   no allow-service ntp
   no allow-service ospf
   no allow-service stun
  !
  no shutdown
 !
!
vpn 512
 interface eth0
  ip address 10.82.83.115/20
  no shutdown
 !
!
```

Step 4 The **Root Certificate or Certificate Chain on the local vEdge** needs to trust the same certificate we installed on the ZTP Server. Symantec certificates are installed on the Lab controllers typically but in this lab we

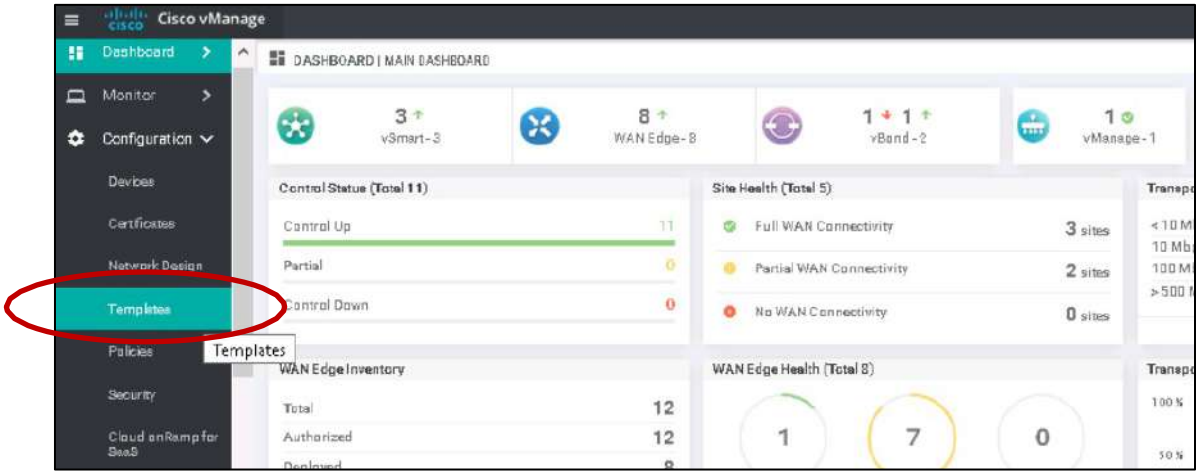
are using a Root CA Server of our own making. We will install the same root-ca-chain certificate we used on the ZTP-Server. To ensure these root-ca-certificates match we will remove the current root-ca-cert.

```
vedge# request root-cert-chain uninstall
Successfully uninstalled the root certificate chain
vedge# request root-cert-chain install
scp://admin@10.82.83.131:/home/admin/ROOTCA.pem vpn 512
Uploading root-ca-cert-chain via VPN 512
Copying ... admin@10.82.83.131:/home/admin/ROOTCA.pem via VPN 512
Warning: Permanently added '10.82.83.131' (ECDSA) to the list of known hosts.
viptela 19.2.097

admin@10.82.83.131's password:<admin>
ROOTCA.pem
100% 1444     3.1MB/s   00:00
Installing the new root certificate chain
Successfully installed the root certificate chain

vedge# show certificate root-ca-cert | inc NterOne
        Issuer: C=US, ST=North Carolina, L=RTP, O=NterOne Lab, OU=NterOne,
CN=NterOne Lab/emailAddress=admin@example.com
        Subject: C=US, ST=North Carolina, L=RTP, O=NterOne Lab, OU=NterOne,
CN=NterOne Lab/emailAddress=admin@example.com
```

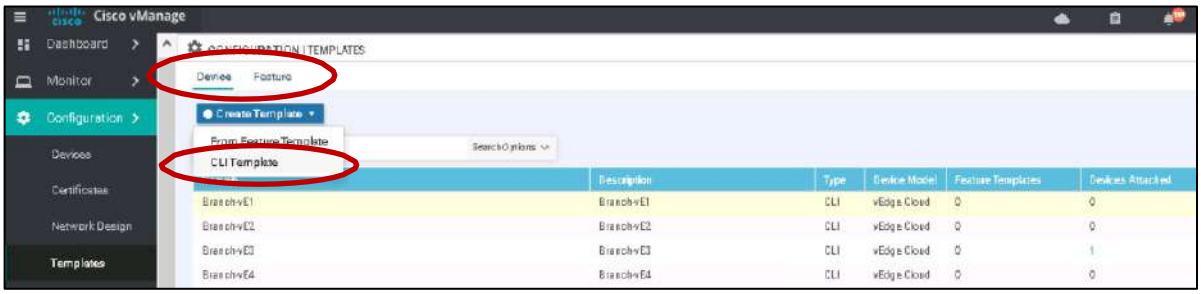
Step 5 Open a browser window and **login to vManage**. You will use vManage to create a **CLI Template** for the vEdge router to be onboarded.

Step 6 Go to **Configuration > Templates.**

 © 2020 NterOne Corporation

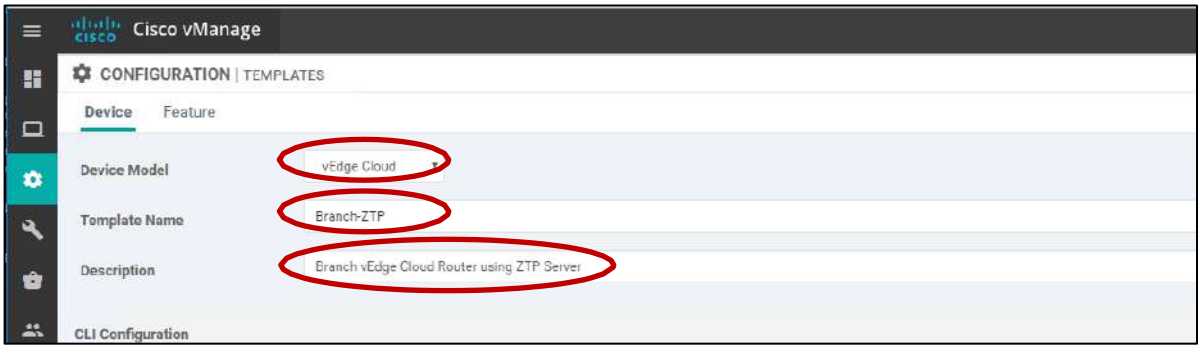


Step 7 Click on **Create Template** and then select **CLI Template** from the drop-down**.**



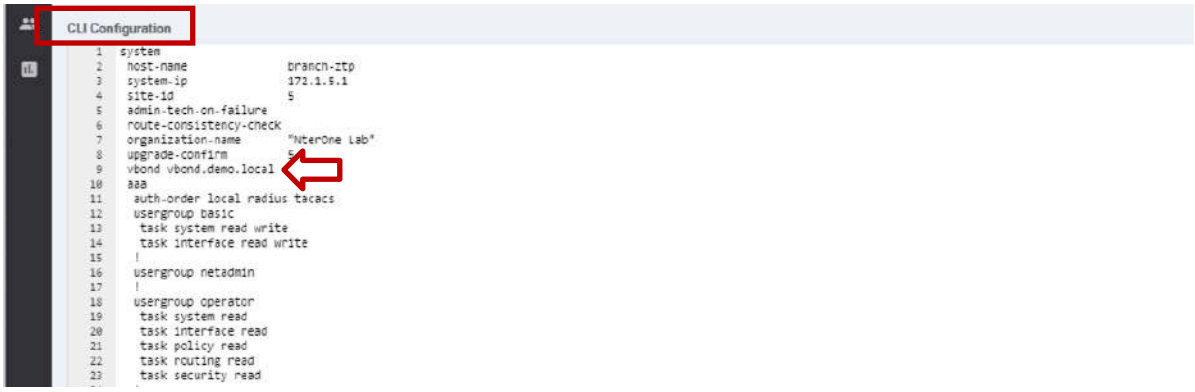
Step 8 On the Templates window select vEdge Cloud and enter the following values.

- Device Model: **vEdge Cloud**
- Template Name: **Branch-ZTP**
- Description: **Branch vEdge Cloud Router using ZTP Server** *(descriptions are required)*



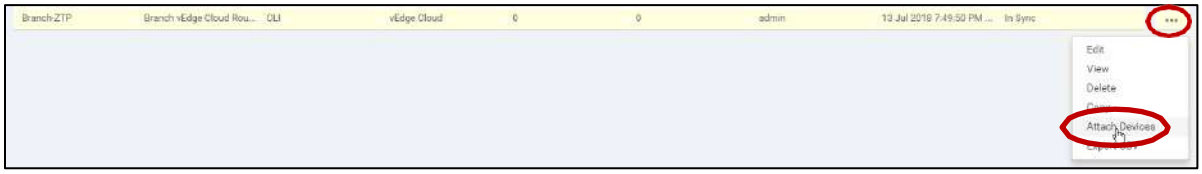
Step 9 **Paste the contents of the Branch-ZTP.txt** file (SDWAN folder on desktop) into the **CLI Configuration** area of the window.

Click Add when done.



Step 10 Verify the Template and modify if necessary.

Step 11 Click the **…** over to the right side of the **Branch-ZTP template** and select **Attach Devices**.

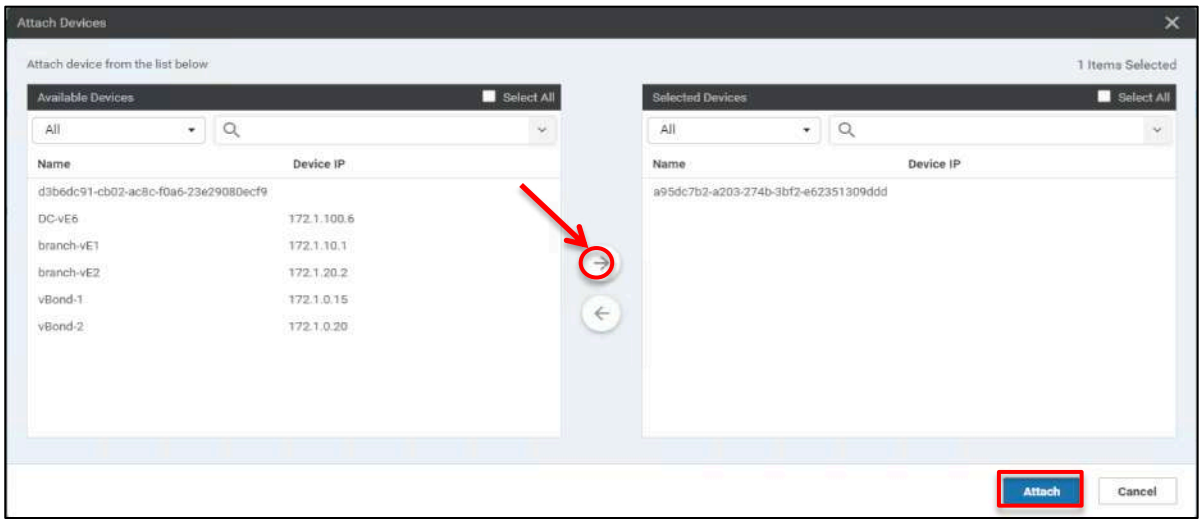


Step 12 The Attach Devices window will appear. Select ONE available/unassigned Chassis Number from the **Available Devices** list and add it to the right under **Selected Devices**. Click **Attach** when done.

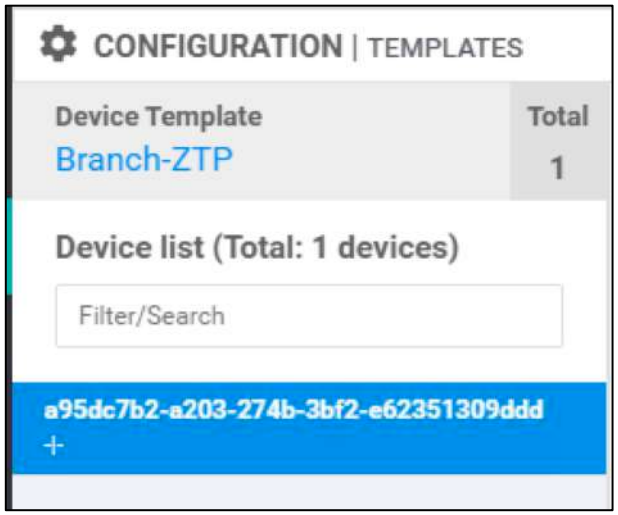
To make it easier you can use the example in the Lab Guide below –

a95dc7b2-a203-274b-3bf2-e62351309ddd

 © 2020 NterOne Corporation



Step 13 On the next screen select the Chassis Number on the Device list.



Note	The chassis number does not match the Branch-ZTP router and this is on purpose and will cause a failure.

Step 14 **Once the device is selected**, then the configuration to be **pushed** will be displayed over to the right-side.

Scroll up or down to verify what **settings will be pushed** to the device.

If there are any **missing features or incorrect values** click **Cancel** and go back and add or modify respectively. Repeat the previous steps to attach the Template if necessary. Verify with the following:

```
viptela-system:system
    device-model            vedge-cloud
```

```
     host-name               branch-ztp
     system-ip               172.1.5.1
     site-id                 5
     admin-tech-on-failure
     route-consistency-check
     sp-organization-name    "NterOne Lab"
     organization-name       “NterOne Lab”
     upgrade-confirm         5
     vbond vbond.demo.local port 12346
     aaa
      auth-order local radius tacacs
      usergroup basic
       task system read write
       task interface read write
      !
      usergroup netadmin
      !
      usergroup operator
       task system read
       task interface read
       task policy read
       task routing read
       task security read
      !
      usergroup tenantadmin
      !
      user admin
       password
$6$lV1XHpFjQrUBVnS1$5zj18TA8WR5oxNGGoScmpUtrL81xFh61qvytKxdkQfrO9n.72
K/j2IiObyz4hLajklMqBe84DRHYQNyXexEta0
      !
     !
     logging
      disk
       enable
      !
     !
    !
    bfd app-route poll-interval 60000
    omp
     no shutdown
     graceful-restart
     advertise ospf external
     advertise connected
     advertise static
    !
    security
     ipsec
      authentication-type ah-sha1-hmac sha1-hmac
     !
    !
    vpn 0
     dns 1.4.28.89 primary
     interface ge0/0
      ip address 9.9.9.169/24
      tunnel-interface
       encapsulation ipsec
       color silver
       no allow-service bgp
       allow-service dhcp
       allow-service dns
       allow-service icmp
       no allow-service sshd
```

 © 2020 NterOne Corporation

```
    no allow-service netconf
    no allow-service ntp
    no allow-service ospf
    no allow-service stun
    allow-service https
   !
   no shutdown
  !
  interface ge0/1
   ip address 1.1.7.169/24
   tunnel-interface
    encapsulation ipsec
    color gold
    no allow-service bgp
    allow-service dhcp
    allow-service dns
    allow-service icmp
    no allow-service sshd
    no allow-service netconf
    no allow-service ntp
    no allow-service ospf
    no allow-service stun
    allow-service https
   !
   no shutdown
  !
  ip route 0.0.0.0/0 1.1.7.22
  ip route 0.0.0.0/0 9.9.9.23
 !
 vpn 1
  router
   ospf
    router-id 172.1.5.1
    timers spf 200 1000 10000
    redistribute static
    redistribute omp
    area 0
     interface ge0/2
     exit
    exit
   !
  !
  interface ge0/2
   ip address 192.168.5.169/24
   no shutdown
  !
 !
 vpn 512
  interface eth0
   ip address 10.82.85.115/20
   no shutdown
  !
 !
 policy
  app-visibility
  flow-visibility
 !
```

Step 15 Click **Configure Devices** to proceed with the Device Template attachment.

Step 16 You are redirected to Task view to monitor the progress. Confirm that it was **scheduled successfully**.

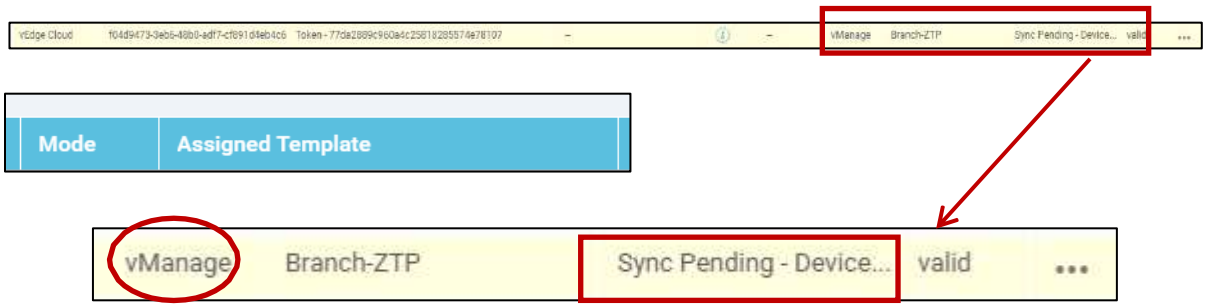
The device is not yet online, but when it does come online, it will have the **Template pushed** to it by vManage.

Step 17 If the **Status** is **Failed** go back and correct whatever necessary or **contact your instructor for assistance**.

Step 18 Go to **Configuration** → **Devices** and verify the entry and its associated **Device Template**.

The Device Mode has now been changed from **CLI to vManage** and the Template has been assigned. When the device is in vManage Mode you cannot make any changes in the CLI of the device.

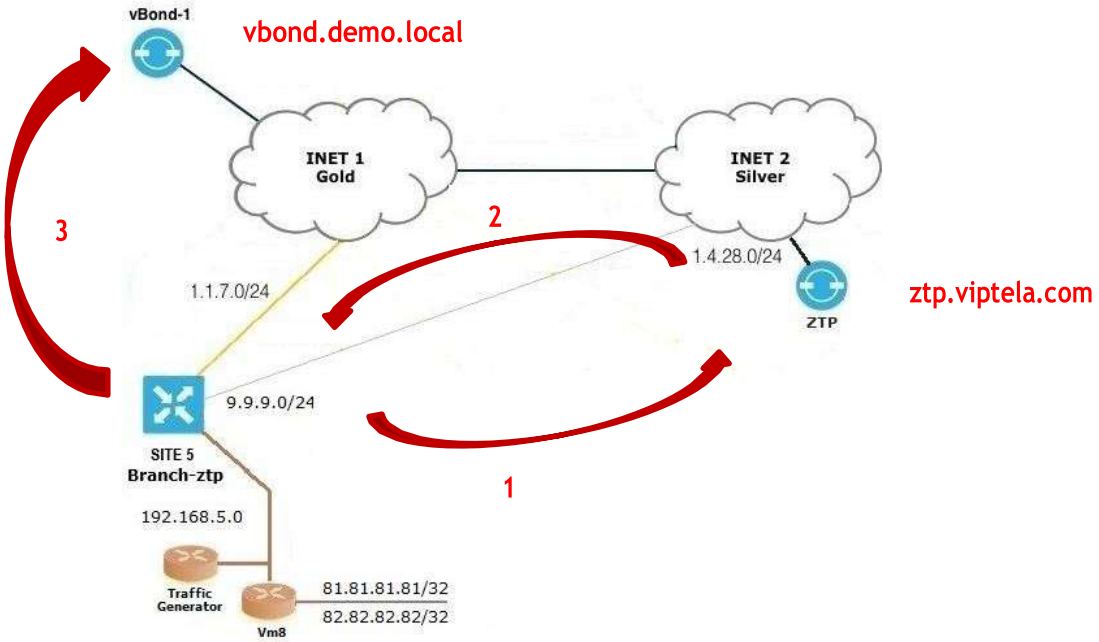
Note Notice that the status of the device is **Sync Pending**. Once the device is online, then vManage will attach the Template to it.



Step 19 Log back into **Branch-ZTP** using SSH. (Refer to the table at the beginning of the lab)

 © 2020 NterOne Corporation

Step 20 We have provided the necessary root-ca-certificate information and values earlier in this task. However, in order to utilize ZTP, Cloud routers require some key values to identify themselves to the vManage/vBond inventory.



Step 21 **During the ZTP process, you will monitor transactions occurring between ZTP, vBond and vManage.**

Step 22 Use the **Linux Shell** on the vEdge router in order to see detailed messages regarding system processes.

Step 23 Use the command **"vshell"** on the CLI to enter the linux shell.

```
vedge# vshell
vedge:~$
```

Step 24 When you enter **vshell** you will be placed into the current users **Home directory**. Use "**pwd**" to verify (print working directory). Use "**ls**" (list files) to view the contents of the directory.

```
vedge:~$ pwd
/home/admin

vedge:~$ ls
CA.crt  archive_id_rsa.pub  root.cer  vedge_cloud_csr
```

Step 25 Change the working directory to **"/var/log/tmplog"**. This is where the **vdebug** file is located. This file will contain the results of any debugs that are enabled. Add **"-l"** to the **"ls"** command to view detailed contents of the directory.

```
vedge:~$ cd /var/log/tmplog
vedge:/var/log/tmplog$
```

```
vedge:/var/log/tmplog$ ls -l (lowercase L)
-rw-r--r-- 1 root root 178171 Jul 1 20:28 vdebug
```

Step 26 Use the command **"cat vdebug"** *(concatenate- cat)* to view the file. Alternatively, you can add **"| more"** to paginate through the contents of the file.

```
vedge:/var/log/tmplog$ cat vdebug

--- OUTPUT OMITTED ---

local7.info: Jul 1 14:32:44 vedge DHCP_CLIENT[800]: %Viptela-
vedge-vdhcpcd-6-INFO- 1400002: Notification: 7/1/2018 14:32:44
dhcp-address-renewed severity-level:minor host- name:"vedge"
system-ip::: vpn-id:0 if-name:"ge0/0" client-
mac:"00:50:56:b9:26:3b" ip:9.9.9.209
local7.debug: Jul 1 16:00:13 vedge stray: [msg] Nameserver 1.4.28.89:53
is back up
local7.info: Jul 1 20:28:01 vedge SYSMGR[277]: %Viptela-vedge-
sysmgrd-6-INFO-1400002: Notification: 7/1/2018 20:28:1 system-
login-change severity-level:minor host-name:"vedge" system-ip:::
user-name:"admin" user-id:37
```

```
vedge:/var/log/tmplog$ cat vdebug | more

--- OUPUT OMITTED ---

local7.debug: Jul 1 12:34:08 vedge vScript: show root cert script
called:
/usr/bin/vconfd_script_show_root_ca_cert.sh -cli
local7.debug: Jul 1 12:34:14 vedge vScript: Sending USR1 signal to
sysmgrd(263)
local7.info: Jul 1 12:34:14 vedge SYSMGR[263]: %Viptela-
vedge-SYSMGR-6-INFO-200040: Received USR1 signal, stopping
watchdog!
local7.info: Jul 1 12:34:14 vedge SYSMGR[263]: %Viptela-vedge-SYSMGR-6-
INFO-200040: wrote
```

 © 2020 NterOne Corporation

```
1 bytes, closing wd 0xb
--More--
```

Step 27 "**Exit**" back to the vEdge CLI and **clear the contents of vdebug using the "request"** command.

```
vedge:/var/log/tmplog$ exit
exit
vedge# request reset logs
Are you sure you want to clear all logs? [yes,NO] yes
vedge#
```

Step 28 **Confirm that interface ge0/0 has received an IP Address via DHCP**. Contact your Instructor if there is no assignment.

```
vedge# show dhcp interface ge0/0
                                                                TIME
VPN   INTERFACE STATE ACQUIRED IP      SERVER   LEASE TIME    REMAINING   GATEWAY   INDEX    DNS
-----------------------------------------------------------------------------------------------
0     ge0/0     bound  9.9.9.202/24   9.9.9.23 1:00:00:00    0:14:46:13  9.9.9.23   0      1.4.28.89
```

Note	**If for any reason DHCP is not working, alternatively configure the interface with static values.**

Step 29 **Open a separate putty window and connect to the ZTP Server using SSH**. *(Refer to the table at the beginning of the lab)* Go to **vshell** and change to the **/var/log/tmplog** directory.

```
ZTP-Server# vshell
ZTP-Server:~$ cd /var/log/tmplog
ZTP-Server:/var/log/tmplog$
```

Step 30 Issue the **"tail -f"** command to actively monitor Log entries added to the **vdebug file**.

```
ZTP-Server:/var/log/tmplog$ tail -f vdebug
```

```
local7.debug: Jul 1 20:55:30 vedge vScript: Chassis number FD8DA2EA-
0C45-4412-AB7A-84CF70C9C7F9 successfully added to the database

local7.debug: Jul 1 20:55:38 vedge vScript: vSmart chassis-numbers add to ztp DB:
/usr/bin/vconfd_script_add_chassis_numbers_to_ztp.sh -cli chassis
50e3cdd0-cde4-4e0f-800d- bab4c288ff15 serial
1eb79482daabe4d8e87bae74c081e545 validity valid vbond
vbond.demo.local org NterOne Lab
```

```
local7.debug: Jul 13 20:55:38 vedge vScript: Chassis number 50E3CDD0-
CDE4-4E0F-800D-BAB4C288FF15 successfully added to the database

local7.info: Jul 13 21:50:56 vedge SYSMGR[271]: %Viptela-vedge-sysmgrd-6-
INFO-1400002: Notification: 7/1/2018 21:50:56 system-logout-change
severity-level:minor host-name:"ZTP-Server" system-ip::: user-
name:"admin" user-id:38

local7.info: Jul 13 22:25:43 vedge SYSMGR[271]: %Viptela-vedge-sysmgrd-6-
INFO-1400002:   Notification:   7/1/2018   22:25:43   system-login-change
severity-level:minor     host-name:"ZTP-Server"     system-ip:::     user-
name:"admin" user-id:40

local7.debug: Jul 13 22:39:10 vedge vScript: show root cert script called:
/usr/bin/vconfd_script_show_root_ca_cert.sh -cli

local7.info: Jul 13 23:15:56 vedge SYSMGR[271]: %Viptela-vedge-sysmgrd-6-
INFO-1400002: Notification: 7/1/2018 23:15:56 system-logout-change
severity-level:minor host-name:"ZTP-Server" system-ip::: user-
name:"admin" user-id:40

local7.info: Jul 14 22:49:49 vedge SYSMGR[271]: %Viptela-vedge-sysmgrd-6-
INFO-1400002:   Notification:   7/1/2018   22:49:49   system-login-change
severity-level:minor     host-name:"ZTP-Server"     system-ip:::     user-
name:"admin" user-id:41

local7.info: Jul 14 23:19:50 vedge SYSMGR[271]: %Viptela-vedge-sysmgrd-6-
INFO-1400002: Notification: 7/1/2018 23:19:50 system-logout-change
severity-level:minor host-name:"ZTP-Server" system-ip::: user-
name:"admin" user-id:41

local7.info: Jul 14 23:31:47 vedge SYSMGR[271]: %Viptela-vedge-sysmgrd-6-
INFO-1400002:   Notification:   7/1/2018   23:31:46   system-login-change
severity-level:minor     host-name:"ZTP-Server"     system-ip:::     user-
name:"admin" user-id:42
```

Step 31 **Open a second putty window to the Branch-ZTP router.**

Step 32 **On the second window run the tail command to actively monitor the vdebug file.**

```
vedge# vshell
vedge:~$ cd /var/log/tmplog
vedge:/var/log/tmplog$ vedge:/var/log/tmplog$ tail -f vdebug

--- OUTPUT OMITTED ---
local7.info: Jul 1 00:03:23 vedge CFGMGR[782]: %Viptela-
vedge-cfgmgr-6-INFO-1400002: Notification: 7/15/2018 0:3:23
system-commit severity-level:minor host-name:"vedge" system-
ip::: user-name:"admin"
```

Step 33 **On the branch-ZTP router window perform the following steps.**

Step 34 Go to **Config mode** and enter the following values into the CLI. **For Chassis and Token number make sure to use the values chosen in the previous task.**

```
vedge# conf
```

 © 2020 NterOne Corporation

```
Entering configuration mode terminal
vedge(config)# system
vedge(config-system)# organization-name "NterOne Lab" (case sensitive)
vedge(config-system)# end
Uncommitted changes found, commit them? [yes/no/CANCEL] yes
Commit complete.
vedge# show run system
system
 host-name               vedge
 admin-tech-on-failure
 no route-consistency-check
 organization-name       "NterOne Lab"
 vbond ztp.viptela.com

--- OUTPUT OMITTED ---
```

Step 35 Use the "**request**" command to **provide the Branch-ZTP router a chassis number and token number** that it will use when establishing its connection to the ZTP Server. **Intentionally enter a wrong value**.

Note	**Create an incorrect entry. Change the first character of the Chassis Number to a different value.**

```
Chassis Number:     f95DC7B2-A203-274B-3BF2-E62351309DDD (device you chose in vManage)
Token Number:       2086AFE3869762E2DBD773B542BA3FC1     (associated token value)

vedge#request vedge-cloud activate chassis-number f95DC7B2-A203-274B-3BF2-E62351309DDD token 2086AFE3869762E2DBD773B542BA3FC1
```

Execute the request command once the change has been made.

Step 36 Use the **"show control local-properties"** command to view the current control setup.

Notice that it has established the **identity of the ZTP Server (1.4.28.169)**.

```
vedge# show control local-properties
personality                 vedge
sp-organization-name        "NterOne Lab"
organization-name           "NterOne Lab"
```

```
certificate-status           Not-Installed
root-ca-chain-status  Installed


certificate-validity       Not       Applicable
certificate-not-valid-before   Not   Applicable
certificate-not-valid-after Not Applicable


dns-name                     ztp.viptela.com
site-id                      0
domain-id                    1
protocol                     dtls
tls-port                     0
system-ip                    0.0.0.0
chassis-num/unique-id        a04d9473-3eb6-48b0-adf7-cf891d4eb4c6
serial-num                   No certificate installed
token                        77da2889c960a4c25818285574e78107
keygen-interval              1:00:00:00
retry-interval               0:00:00:19
no-activity-exp-interval     0:00:00:12
dns-cache-ttl                0:00:02:00
port-hopped                  FALSE
time-since-last-port-hop     0:00:00:00
number-vbond-peers           1

INDEX   IP                                      PORT
-----------------------------------------------------
0       1.4.28.169                              12346

number-active-wan-interfaces 1


 --- LEGEND OMITTED  ---
            PUBLIC          PUBLIC PRIVATE         PRIVATE
 PRIVATE                              MAX   RESTRICT/           LAST         SPI TIME
 NAT  VM
 INTERFACE  IPv4            PORT   IPv4             IPv6
 PORT    VS/VM COLOR            STATE CNTRL  CONTROL/    LR/LB  CONNECTION   REMAINING
TYPE CON


STUN                                                          PRF
-----------------------------------------------------------------------------------------
ge0/0   9.9.9.209        12346   9.9.9.209        ::
12346   0/0         default       up      2     no/yes/no    No/No        0:00:00:02
        0:00:00:00  N
```

 © 2020 NterOne Corporation

5

Step 37 **On the ZTP Server** you should notice some Log entries corresponding to the **vEdge routers attempt to connect** to the Server.

Recall that the ZTP server is a top-level vBond orchestrator and it will verify if the device has been entered into the ZTP database. **If the device is found in the database**, then it will provide the vEdge router with the organizations **assigned vBond orchestrator**.

```
local7.debug: Jul 1 00:33:48 vedge VBOND[20164]: vbond_handshake_event_cb[5749]:
%VDAEMON_DBG_MISC-1:
Get CA RSA Public key

local7.debug: Jul 1 00:33:48 vedge VBOND[20164]: vbond_proc_challenge_ack[3816]:
%VDAEMON_DBG_MISC-1:
No signed challenge sent by the Peer.. Sent OTP instead

local7.debug: Jul 1 00:33:48 vedge VBOND[20164]:
vdaemon_find_vedge_in_vedge_list[161]:
%VDAEMON_DBG_MISC-1: Peer's device (A04D9473-3EB6-48B0-ADF7-CF891D4EB4C6) not
found...

local7.info: Jul 1 00:33:48 vedge VBOND[20164]: %Viptela-ZTP-Server-
vbond_0-6-INFO-1400002: Notification: 7/1/2018 0:33:48 vbond-reject-
vedge-connection severity-level:major host-name:"ZTP- Server" system-
ip::: uuid:"A04D9473-3EB6-48B0-ADF7-CF891D4EB4C6" organization-
name:"NterOne Lab" sp- organization-name:"NterOne Lab"
reason:"ERR_SER_NUM_NT_PRESENT"
```

Step 38 Enter **Ctrl-C to cancel the tail**, and then use **"grep"** to filter the vdebug file. (global regular expression print – grep)(-E is used to list an expression for the search value)

```
em-ip::: local-system-ip:0.0.0.0 local-color:default
reason:"ERR_SER_NUM_NT_PRESENT"
^C

ZTP-Server:/var/log/tmplog$ grep -E 'local7.debug:.*VBOND' vdebug

--- OUTPUT OMITTED ---

local7.debug: Jul 1 01:03:45 vedge VBOND[20164]:
vdaemon_find_vedge_in_vedge_list[161]:
%VDAEMON_DBG_MISC-1: Peer's device (A04D9473-3EB6-48B0-ADF7-CF891D4EB4C6) not
found...

local7.debug: Jul 1 01:04:02 vedge VBOND[20164]: vbond_handshake_event_cb[5749]:
%VDAEMON_DBG_MISC-1: Get CA RSA Public key

local7.debug: Jul 1 01:04:02 vedge VBOND[20164]: vbond_proc_challenge_ack[3816]:
%VDAEMON_DBG_MISC-1: No signed challenge sent by the Peer.. Sent OTP instead

local7.debug: Jul 1 01:04:02 vedge VBOND[20164]:
vdaemon_find_vedge_in_vedge_list[161]:
%VDAEMON_DBG_MISC-1: Peer's device (A04D9473-3EB6-48B0-ADF7-CF891D4EB4C6) not
found...
```

```
local7.debug: Jul 1 01:04:19 vedge VBOND[20164]: vbond_handshake_event_cb[5749]:
%VDAEMON_DBG_MISC-1: Get CA RSA Public key

local7.debug: Jul 1 01:04:19 vedge VBOND[20164]: vbond_proc_challenge_ack[3816]:
%VDAEMON_DBG_MISC-1: No signed challenge sent by the Peer.. Sent OTP instead

local7.debug: Jul 1 01:04:19 vedge VBOND[20164]:
vdaemon_find_vedge_in_vedge_list[161]:
%VDAEMON_DBG_MISC-1: Peer's device (A04D9473-3EB6-48B0-ADF7-CF891D4EB4C6) not
found...
```

Step 39 The previous command filtered for the keywords **"local7.debug"** AND **"VBOND"** by using **.*** between the keywords. Notice the vbond `vbond_handshake_event_cb` and `vbond_proc_challenge_ack` messages. Since a vEdge Cloud router is used here, there is no certificate present, so instead it uses the Token number (OTP) for the challenge phase.

Step 40 Use the same command from before, but this time use **grep to filter the output twice** to search for a **specific Device in the Log file**.

Notice that the Device cannot be found in the vedge list - `vdaemon_find_vedge_in_vedge_list`

Grep is case-sensitive by default. Use "–i" to ignore case The command and all values are entered on 1 line.

```
ZTP-Server:/var/log/tmplog$ grep -E 'local7.debug:.*VBOND'
vdebug | grep "A04D9473-3EB6- 48B0-ADF7-CF891D4EB4C6"
```

OR to ignore case use "-i" *(lowercase i)*

```
ZTP-Server:/var/log/tmplog$ grep -i -E 'LOcal7.DEBUG:.*VbonD'
vdebug | grep -i "A04D9473- 3EB6-48B0-ADF7-CF891D4EB4C6"
```

```
--- OUTPUT OMITTED ---

local7.debug: Jul 1 00:57:39 vedge VBOND[20164]:
vdaemon_find_vedge_in_vedge_list[161]:
   %VDAEMON_DBG_MISC-1: Peer's device (A04D9473-3EB6-48B0-ADF7-
   CF891D4EB4C6) not found...
```

Step 41 Modify the previous command to search for **"local7.info"** entries instead of debug entries. Notice it clearly indicates the **Device Serial number is not present in the ZTP Server**.

```
ZTP-Server:/var/log/tmplog$ grep -E 'local7.info:.*VBOND'
vdebug | grep "A04D9473-3EB6- 48B0-ADF7-CF891D4EB4C6"
```

```
--- OUTPUT OMITTED ---
```

 © 2020 NterOne Corporation

```
local7.info: Jul 1 01:42:04 vedge VBOND[20164]: %Viptela-ZTP-
Server-vbond_0-6-INFO- 1400002: Notification: 7/1/2018 1:42:4
vbond-reject-vedge-connection severity-level:major host-
name:"ZTP-Server" system-ip::: uuid:"A04D9473-3EB6-48B0-ADF7-
CF891D4EB4C6" organization-name:"NterOne Lab" sp-organization-
name:"NterOne Lab" reason:"ERR_SER_NUM_NT_PRESENT"
```

Step 42 **Go back to the second Branch-ZTP putty window and view the ongoing tail.**

```
local7.debug: Jul 1 02:48:53 vedge VDAEMON[799]:
vdaemon_find_next_active_wan_intf[1423]:
%VDAEMON_DBG_MISC-1: Next wan interface to connect to vmanage = ge0_0

local7.debug: Jul 1 02:48:53 vedge stray: setsockopt: Bad file descriptor

local7.debug: Jul 1 02:48:53 vedge VDAEMON[799]:
vdaemon_set_confd_ztp_status[6294]:
%VDAEMON_DBG_MISC-1: Setting ztp-status to 0

local7.debug: Jul 1 02:48:53 vedge VDAEMON[799]:
vdaemon_cfg_confd_params_threaded_set[5748]: %VDAEMON_DBG_CONFD-1: Setting ztp
status to 0

local7.debug: Jul 1 02:48:53 vedge VDAEMON[799]:
vdaemon_cfg_confd_params_threaded_set[5770]: %VDAEMON_DBG_CONFD-1: ztp status = 0

local7.debug: Jul 1 02:48:53 vedge VDAEMON[799]:
vdaemon_cfg_confd_params_threaded_set[6160]: %VDAEMON_DBG_MISC-1: Applying
configuration after 0 retries to 0..

local7.debug: Jul 1 02:48:54 vedge VDAEMON[799]:
vdaemon_find_next_active_wan_intf[1423]:
%VDAEMON_DBG_MISC-1: Next wan interface to connect to vmanage = none
```

Step 43 Enter **Ctrl-C to cancel the tail**, and then use **"grep"** to filter the vdebug file. The **"–A 4"** indicates it will **print the next 4 lines after** the value(s) has been found in the log file.

```
local7.debug: Jul 1 02:58:48 vedge VDAEMON[799]:
vdaemon_find_next_active_wan_intf[1423]:
%VDAEMON_DBG_MISC-1: Next wan interface to connect to vmanage = none
^C
```

```
vedge:/var/log/tmplog$ grep -E 'local7.debug.*VDAEMON' vdebug | grep -A 4 "Next
wan interface to connect to vmanage = ge0_0"


local7.debug: Jul 1 03:12:59 vedge VDAEMON[799]:
vdaemon_find_next_active_wan_intf[1423]:

%VDAEMON_DBG_MISC-1: Next wan interface to connect to vmanage = ge0_0

local7.debug: Jul 1 03:12:59 vedge VDAEMON[799]:
vdaemon_set_confd_ztp_status[6294]:
%VDAEMON_DBG_MISC-1: Setting ztp-status to 0

local7.debug: Jul 1 03:12:59 vedge VDAEMON[799]:
vdaemon_cfg_confd_params_threaded_set[5748]:%VDAEMON_DBG_CONFD-1: Setting ztp
status to 0

local7.debug: Jul 1 03:12:59 vedge VDAEMON[799]:
vdaemon_cfg_confd_params_threaded_set[5770]:%VDAEMON_DBG_CONFD-1:   ztp status = 0

local7.debug: Jul 1 03:12:59 vedge VDAEMON[799]:
vdaemon_cfg_confd_params_threaded_set[6160]:%VDAEMON_DBG_MISC-1: Applying
configuration after 0 retries to 0..
```

4 Lines

Step 44 From the output of the previous command it appears that the **Branch-ZTP router is making an attempt to connect to the ZTP Server using ge0/0**, but fails to confirm connectivity. **Unfortunately it doesn't specify why** it's failing to connect to the ZTP Server.

Step 45 **On the first Branch-ZTP router window** use the **"debug"** command to log additional info.

```
vedge# debug vdaemon misc high
```

Step 46 **Wait a moment or two, and then on the second Branch-ZTP router window use the following grep commands in vshell.**

```
vedge:/var/log/tmplog$ grep -E 'local7.debug.*VDAEMON' vdebug
| grep "vdaemon_ftm_send_ctrl_tun"


local7.debug: Jul 1 04:10:49 vedge VDAEMON[799]:
vdaemon_ftm_send_ctrl_tun[241]:
%VDAEMON_DBG_MISC-3: Local-TLOC 9.9.9.209:12346 Remote-TLOC
1.4.28.169:12346 msg-type
Delete


vedge:/var/log/tmplog$ grep -E 'local7.debug.*VDAEMON' vdebug | grep
"vbond_parse"


local7.debug: Jul 1 04:10:38 vedge VDAEMON[799]:
vbond_parse_challenge[5684]:
%VDAEMON_DBG_MISC-3: Parsing CHALLENGE ..

local7.debug: Jul 1 04:10:38 vedge VDAEMON[799]: vbond_parse_msg[2011]:
%VDAEMON_DBG_MISC-3: Received a TEAR DOWN from the peer
```

repeats

 © 2020 NterOne Corporation

```
vedge:/var/log/tmplog$ grep -E 'local7.debug.*VDAEMON' vdebug | grep
"vdaemon_send_challenge_ack"

local7.debug: Jul 1 04:10:55 vedge VDAEMON[799]:
vdaemon_send_challenge_ack[4247]:
%VDAEMON_DBG_MISC-3: Send Challenge ACK ... (board_id_present No)
```

Step 47 It appears that the Branch-ZTP router cannot establish a connection with the ZTP Server due to the fact that the board_id or Chassis number is not found in the ZTP Servers database (vedge list).

Step 48 **Open yet another putty window and connect to vManage using SSH.** *(Refer to the table at the beginning of the lab)*

Step 49 On vManage go to vshell and start a tail on vdebug, but specifically filter for the Branch-ZTP chassis number.

Note Use the correct chassis number and not the wrong value from previous steps

```
vmanage1# vshell
vmanage1:~$ cd /var/log/tmplog
vmanage1:/var/log/tmplog$

vmanage1:/var/log/tmplog$ tail -f vdebug | grep "f95DC7B2-A203-274B-3BF2-
E62351309DDD"
```

Step 50 **On the second Branch-ZTP router window change the directory to "var/log/confd". If a tail is running, then cancel it. List the contents of the directory and take note that there is no netconf.trace file.**

```
vedge# vshell
vedge:~$ cd /var/log/confd
vedge:/var/log/confd$ ls -l

 -rw-r--r--    1   root   root    0   Jul   1      02:32 devel.log
 -rw-r--r--    1   root   root    8   Jul   1      14:30 error.log.1
 -rw-r--r--    1   root   root   18   Jul   1      14:30 error.log.idx
 -rw-r--r--    1   root   root   13   Jul   1      14:30 error.log.siz
 -rw-r--r--    1   root   root    0   Jul   1      02:32 snmp.log
```

Step 51 **Netconf.trace is used to store the XML content** that is exchanged between vEdge and the vManage NMS. This content will contain any necessary configuration exchanges.

Step 52 **On the first Branch-ZTP router window use the debug command to log netconf activity.**

```
vedge# debug netconf traces
```

Step 53 **On the second Branch-ZTP router window view the contents of the directory again. Since debug for netconf has been configured, this time there should be a netconf.trace file present in the directory.**

```
vedge:/var/log/confd$ ls -l
```

```
-rw-r--r-- 1 root root  0 Jul 1 02:32 devel.log
-rw-r--r-- 1 root root  8 Jul 1 14:30 error.log.1
-rw-r--r-- 1 root root 18 Jul 1 14:30 error.log.idx
-rw-r--r-- 1 root root 13 Jul 1 14:30 error.log.siz
-rw-r--r-- 1 root root  0 Jul 1 18:03 netconf.trace
-rw-r--r-- 1 root root  0 Jul 1 02:32 snmp.log
```

Step 54 **The netconf.trace file is currently empty** due to the fact there have been no configuration exchanges as of yet.

```
-rw-r--r-- 1 root root  0 Jul 1 18:03 netconf.trace
```

Step 55 **Go to the first Branch-ZTP router window and re-issue the <u>corrected "request" command.</u>**

```
a95DC7B2-A203-274B-3BF2-E62351309DDD
```

```
Chassis Number:     a95DC7B2-A203-274B-3BF2-E62351309DDD
Token Number:       2086AFE3869762E2DBD773B542BA3FC1     (associated token value)

vedge# request vedge-cloud activate chassis-number A95DC7B2-A203-274B-3BF2-E62351309DDD
    token 2086AFE3869762E2DBD773B542BA3FC1
```

Step 56 **Wait a couple of minutes** while connectivity is being established.

Step 57 Use the following grep commands to **verify ZTP, vManage and vBond activity.**

On vManage (vshell) view the ongoing tail and notice the newly added device.

```
vmanage1:/var/log/tmplog$ tail -f vdebug | grep "A95DC7B2-A203-274B-3BF2-E62351309DDD"
```

 © 2020 NterOne Corporation

```
local7.debug: Jul 1 12:32:11 vManage-1 VDAEMON_4[3455]:
internal_security_add_board_id[944]: %VDAEMON_DBG_MISC-1: Adding
... |bfb0fd2d|F04D9473- 3EB6-48B0-ADF7-CF891D4EB4C6|NterOne Lab|
```

On the ZTP Server (vshell) use the following grep commands to verify successful connectivity.

Notice the successful addition of the device to the ZTP database.

```
ZTP-Server:/var/log/tmplog$ grep "local7.debug:" vdebug | grep
"F04D9473-3EB6-48B0-ADF7- CF891D4EB4C6"
local7.debug: Jul 1 20:13:20 vedge vScript: vSmart chassis-numbers add
to ztp DB:
/usr/bin/vconfd_script_add_chassis_numbers_to_ztp.sh -cli
chassis F04D9473-3EB6-48B0- ADF7-CF891D4EB4C6 serial
77DA2889C960A4C25818285574E78107 validity valid vbond
vbond.demo.local org NterOne Lab
local7.debug: Jul 1 20:13:21 vedge vScript: Chassis number
F04D9473-3EB6-48B0-ADF7- CF891D4EB4C6 successfully added to
the database
ZTP-Server:/var/log/tmplog$
```

Alternatively, you could just filter for the Chassis number by itself.

```
ZTP-Server:/var/log/tmplog$ grep "F04D9473-3EB6-48B0-ADF7-CF891D4EB4C6"
vdebug
```

```
local7.debug: Jul 1 20:13:20 vedge vScript: vSmart chassis-numbers add
to ztp DB:
/usr/bin/vconfd_script_add_chassis_numbers_to_ztp.sh -cli
chassis F04D9473-3EB6-48B0- ADF7-CF891D4EB4C6 serial
77DA2889C960A4C25818285574E78107 validity valid vbond
vbond.demo.local org NterOne Lab
local7.debug: Jul 1 20:13:21 vedge vScript: Chassis number
F04D9473-3EB6-48B0-ADF7- CF891D4EB4C6 successfully added to
the database
ZTP-Server:/var/log/tmplog$
```

On the Branch-ZTP router use the following grep commands to verify successful connectivity.

```
vedge:/var/log/tmplog$ grep -E 'local7.debug.*VDAEMON'
vdebug | grep -A 4 "Next wan interface to connect to
vmanage = ge0_0"
```

The Branch-ZTP router now has receipt of the organization's vBond orchestrator.

```
local7.debug: Jul 2 12:34:24 vedge VDAEMON[799]:
vdaemon_find_next_active_wan_intf[1423]:
%VDAEMON_DBG_MISC-1: Next wan interface to connect to vmanage = ge0_0

local7.debug: Jul 2 12:34:24 vedge VDAEMON[799]: vdaemon_enable_my_tloc[1473]:
%VDAEMON_DBG_MISC-1: Highest preferred interface ge0_0

local7.debug: Jul 2 12:34:24 vedge VDAEMON[799]:
vbond_connect_to_peer_vbonds[7120]:
%VDAEMON_DBG_MISC-3: Connecting to peer vbonds for wan if 'ge0_0'..

local7.debug: Jul 2 12:34:24 vedge VDAEMON[799]:
vbond_connect_to_peer_vbonds[7208]:
%VDAEMON_DBG_MISC-3: Requesting vBond DNS resolution for 'vbond.demo.local' on
ge0_0..

local7.debug: Jul 2 12:34:24 vedge VDAEMON[799]: vdaemon_config_vpn_if[5626]:
%VDAEMON_DBG_CONFD-1: Interface ge0_1 ip address changed
```

```
branch-ztp:/var/log/tmplog$ grep -E 'local7.debug.*VDAEMON' vdebug | grep
"vbond_parse" | more
```

--- Notice a collection of "vbond_parse" messages not seen before ---

```
Branch-ztp:/var/log/tmplog$ grep -E 'local7.debug.*VDAEMON' vdebug | grep -A 7
"Received a NEW register reply message"

local7.debug: Jul 2 12:36:58 vedge VDAEMON[799]: vbond_parse_msg[2054]:
%VDAEMON_DBG_MISC-3: Received a NEW register reply message

local7.debug: Jul 2 12:36:58 vedge VDAEMON[799]:
vbond_parse_new_register_reply[1203]:
%VDAEMON_DBG_MISC-3: serial 06D558D8F3630DCB1B81404B9D072529 org NterOne Lab

local7.debug: Jul 2 12:36:58 vedge VDAEMON[799]:
vbond_parse_new_register_reply[1205]:
%VDAEMON_DBG_MISC-3: cp 0x7fff28af48d0 pktlen 432

local7.debug: Jul 2 12:36:58 vedge VDAEMON[799]:
vbond_parse_new_register_reply[1203]:
%VDAEMON_DBG_MISC-3: serial 07F4E30F8E9368C48C06601C5AB51980 org NterOne Lab

local7.debug: Jul 2 12:36:58 vedge VDAEMON[799]:
vbond_parse_new_register_reply[1205]:
%VDAEMON_DBG_MISC-3: cp 0x7fff28af4950 pktlen 304

local7.debug: Jul 2 12:36:58 vedge VDAEMON[799]:
vbond_parse_new_register_reply[1203]:
%VDAEMON_DBG_MISC-3: serial 1AA04C7DE9392B380677A0E7D1EFB055 org NterOne Lab

local7.debug: Jul 2 12:36:58 vedge VDAEMON[799]:
vbond_parse_new_register_reply[1205]:
%VDAEMON_DBG_MISC-3: cp 0x7fff28af49d0 pktlen 176

local7.debug: Jul 2 12:36:58 vedge VDAEMON[799]:
vbond_parse_new_register_reply[1203]:
%VDAEMON_DBG_MISC-3: serial 69DE12CE976BC070D5184152E0298516 org NterOne Lab
```

7 Lines

The Branch-ZTP router received the controller information from vBond, which includes all vSmart and vManage Serial numbers (certificate).

Step 58 Use the **"show control local-properties"** command to view the current control setup.

 © 2020 NterOne Corporation

A certificate has been installed by vManage and all WAN interfaces are fully functional due to the successful attachment of the Device CLI Template.

```
branch-ztp# show control local-properties
personality                             vedge
sp-organization-name                    "NterOne Lab"
organization-name                       "NterOne Lab"
certificate-status                      Installed
root-ca-chain-status                    Installed


certificate-validity                    Valid
certificate-not-valid-before Jul 2 12:32:05 2018 GMT
certificate-not-valid-after             Jul 1 12:32:05 2028 GMT


dns-name                                vbond.demo.local
site-id                                 5
domain-id                               1
protocol                                dtls
tls-port                                0
system-ip                               172.1.5.1
chassis-num/unique-id                   a95dc7b2-a203-274b-3bf2-e62351309ddd
serial-num                              BFB0FD2D
token                                   Invalid
keygen-interval                         1:00:00:00
retry-interval                          0:00:00:19
no-activity-exp-interval                0:00:00:12
dns-cache-ttl                           0:00:02:00
port-hopped                             TRUE
time-since-last-port-hop                0:00:29:37
number-vbond-peers                      0
number-active-wan-interfaces 2


--- LEGEND OMITTED ---


            PUBLIC           PUBLIC PRIVATE          PRIVATE
 PRIVATE                              MAX    RESTRICT/           LAST          SPI TIME
 NAT   VM
 INTERFACE  IPv4             PORT    IPv4             IPv6
 PORT     VS/VM COLOR             STATE CNTRL CONTROL/     LR/LB  CONNECTION    REMAINING
TYPE CON


STUN                                                          PRF
```

```
--------------------------------------------------------------------------------
ge0/0   9.9.9.169     12386   9.9.9.169   ::
12386   2/1         silver    up          2no/yes/no   No/No   0:00:28:30 0:11:30:45
        E
5
ge0/1   1.1.7.169     12366   1.1.7.169   ::
12366   2/0         gold      up          2no/yes/no   No/No   0:00:28:32 0:11:29:15
        E
5
```

Step 59 **On the Branch-ZTP router go to vshell and change to the " cd /var/log/confd" directory and view the contents of the netconf.trace log file.**

```
branch-ztp:/var/log/confd$ cat netconf.trace | more


**> sess:2568 new session


2-Jul-2018::12:32:04.508 **< sess:2568 write:
<hello xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <capabilities>
    <capability>urn:ietf:params:netconf:base:1.0</capability>
--- OUTPUT OMITTED ---
```

Use the Chassis number to search for configuration values specific to the Branch-ZTP router.

```
branch-ztp:/var/log/confd$ grep -A 300 "<chassis-number>
a95dc7b2-a203-274b-3bf2-e62351309ddd</chassis-number>"
netconf.trace | more


     <chassis-number> a95dc7b2-a203-274b-3bf2-e62351309ddd</chassis-
     number>
      <host-name>vedge</host-name>
      <system-ip>172.1.5.1</system-ip>
      <domain-id>1</domain-id>
--- OUTPUT OMITTED ---
```

Use the "Branch-ZTP" Template name to search the contents of the netconf.trace file.

```
branch-ztp:/var/log/confd$ grep -A 200 "Branch-ZTP" netconf.trace | more
```

 © 2020 NterOne Corporation

```
   <config-template-name>Branch-ZTP</config-template-name>
        <route-consistency-check>true</route-consistency-check>
        <dpi-cache-size nc:operation="remove"
xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"/>
        <port-offset nc:operation="remove"
xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"/>
        <upgrade-confirm>5</upgrade-confirm>
--- OUTPUT OMITTED ---
```

On the vBond1 orchestrator, go to vshell ("cd /var/log/tmplog") and use the following grep command to verify successful addition.

Search for Branch-ZTP routers Chassis number. Make sure to use uppercase when searching or use **–i** to ignore case.

```
vBond-1:/var/log/tmplog$ grep -E 'local7.debug.*VBOND' vdebug |
grep " a95dc7b2-a203-274b-3bf2-e62351309ddd"
```

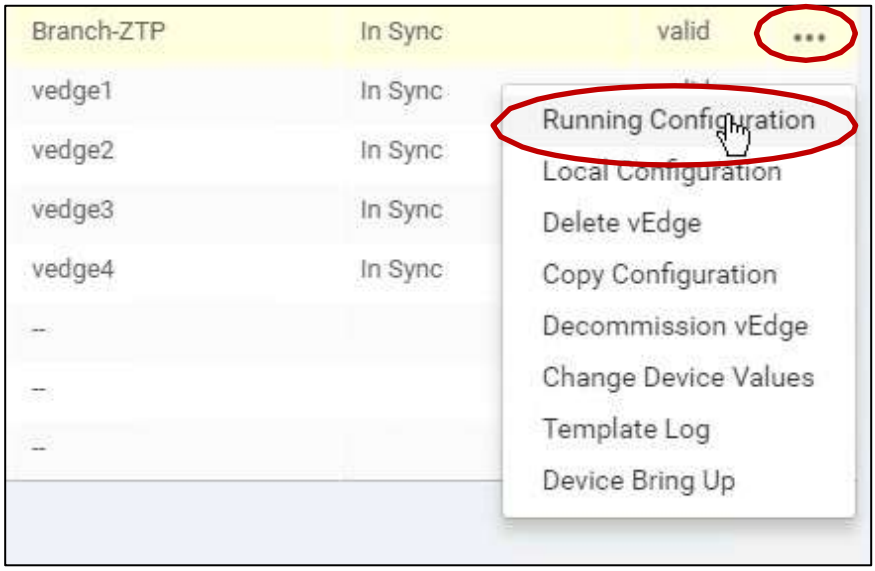
```
local7.debug: Jul 2 12:32:13 vBond-1 VBOND[2698]:
internal_security_add_board_id[944]:
%VDAEMON_DBG_MISC-1: Adding ... |bfb0fd2d| a95dc7b2-a203-274b-3bf2-
e62351309ddd|NterOne Lab|
vBond-1:/var/log/tmplog$
```

Go to the vManage GUI and navigate to Configuration → Templates → Branch-ZTP.

Step 60 **The template should be attached and in Sync with the Branch-ZTP router.**

Step 61 Go to **Configuration → Devices** and view the **Branch-ZTP** entry from the table.

Step 62 Click the **…** to the right of the entry and in the drop-down menu **select running-configuration**



Step 63 You should see the **Running-configuration** of the Branch-ZTP router.

Step 64 End of Task.

 © 2020 NterOne Corporation

Lab 14: Using the REST API and Python Scripting

In this lab, you will install and go thru a basic primer on the REST API and Python.

Activity Procedure

The following devices and values will be used in this Lab.

Device Name	IP Address or URL	Username	Password
vManage-1	https://1.4.28.28 – SSH 10.82.83.131	admin	admin
Branch-vE1	1.1.7.11 – SSH 10.82.83.111	admin	admin

Task 1: Python Installation and Introduction

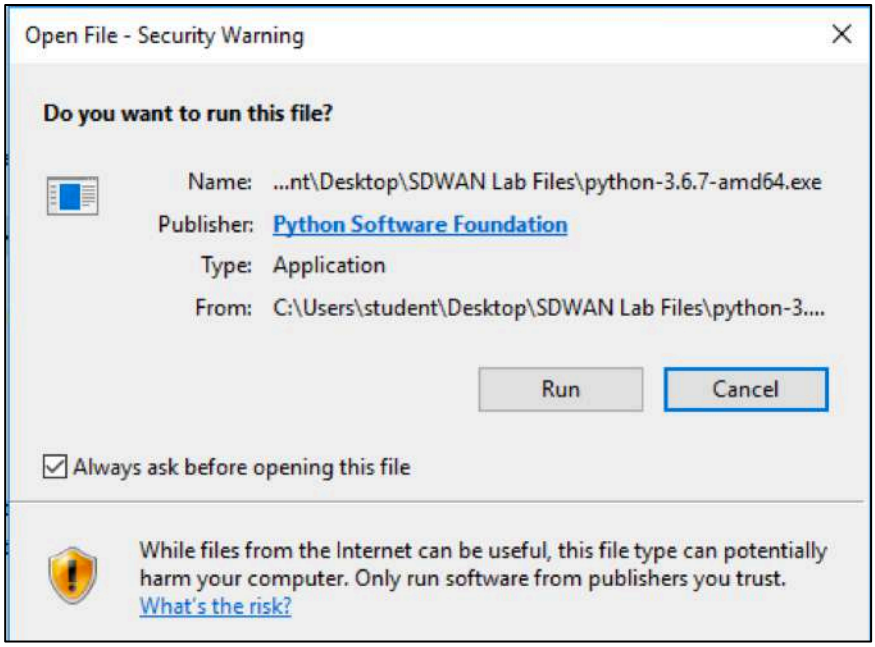
In this task, you will install Python on your admin PC and learn some very basic skills about python.

Activity Procedure

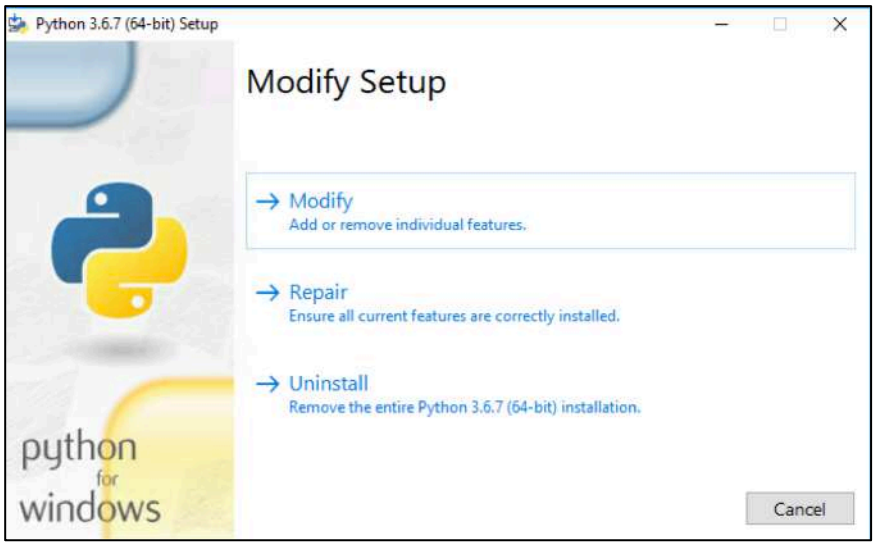
Complete these steps:

Step 1 From the remote desktop of your admin PC, open the SDWAN Lab Files folder on your desktop.

Step 2 Right-click the installer filename **python-3.6.7-amd64.exe** and run as admistrator. Check with the instructor for the administrator password . If Python is already installed , you may skip the installation or you may choose to modify.

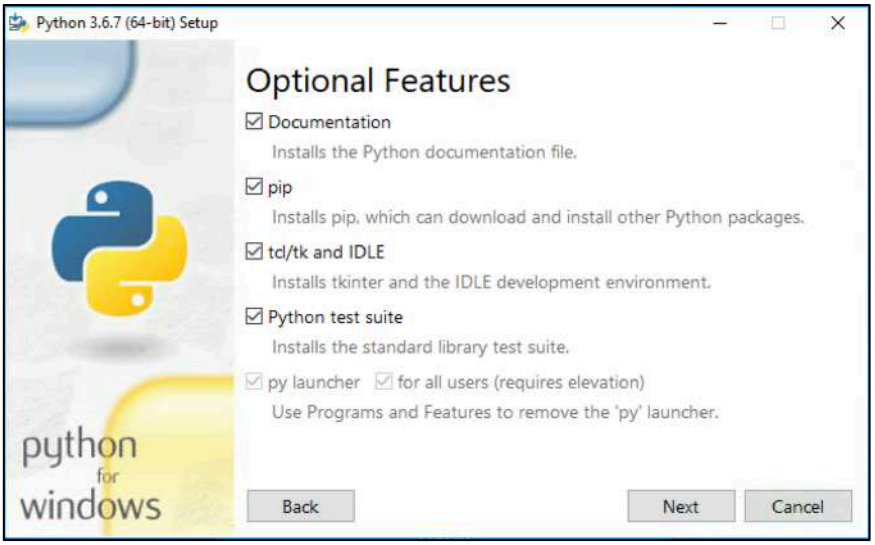


Step 3 On the first window of the installation select the **Modify** option, if already installed; otherwise, select Install.



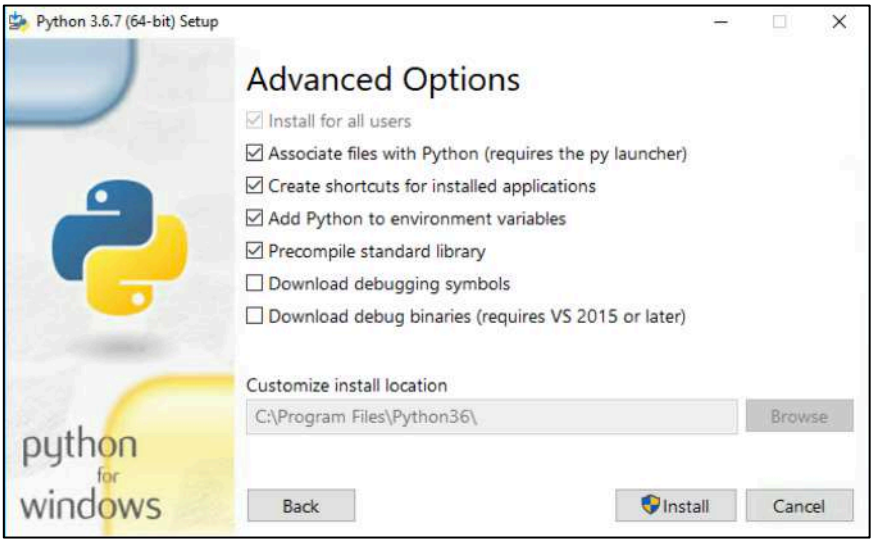
Step 4 Ensure that on the Optional Features page all options are selected, then click **Next**.

 © 2020 NterOne Corporation

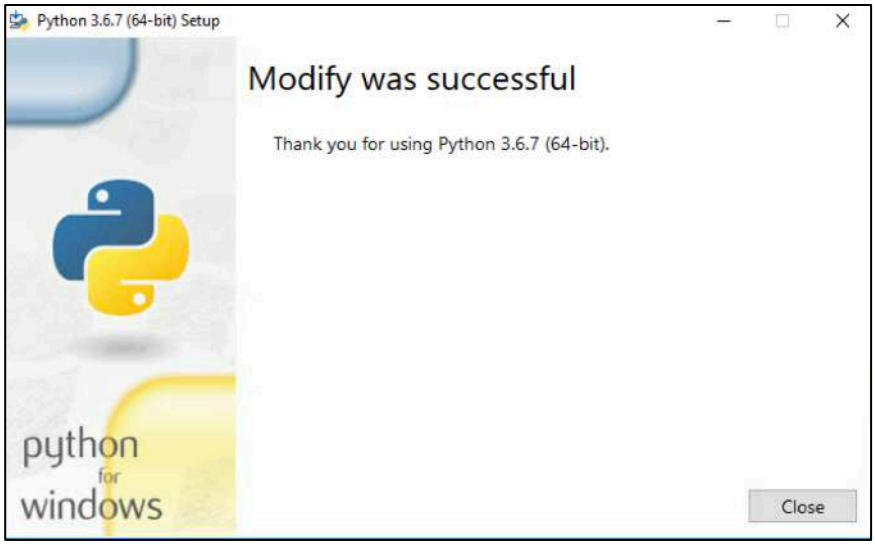


Step 5 On the Advanced Options window check the following options and then press **Install**.

- Install for all users
- Add Python to environment variables



Step 6 When prompted to allow the installation to make changes please select **Yes.** Click Close when the installation has completed.



Step 7 Open a command prompt and run as administrator. Execute the command **pip install requests**. This module will be used in the next task. You may need to be in the correct directory execute the below commands.

c:\users\Administrator\AppData\Local\Programs\Python\Python36\Scripts

```
C:\Users\student>pip install requests
Requirement already satisfied: requests in c:\program files\python36\lib\site-packages (2.21.0)
Requirement already satisfied: certifi>=2017.4.17 in c:\program files\python36\lib\site-packages (from requests) (2018.11.29)
Requirement already satisfied: chardet<3.1.0,>=3.0.2 in c:\program files\python36\lib\site-packages (from requests) (3.0.4)
Requirement already satisfied: urllib3<1.25,>=1.21.1 in c:\program files\python36\lib\site-packages (from requests) (1.24.1)
Requirement already satisfied: idna<2.9,>=2.5 in c:\program files\python36\lib\site-packages (from requests) (2.8)
You are using pip version 19.0.3, however version 19.3 is available.
You should consider upgrading via the 'python -m pip install --upgrade pip' command.
```

Step 8 Run the command **pip install configparser**. This module will be required in the next task.

```
C:\Users\student>pip install configparser
Requirement already satisfied: configparser in c:\program files\python36\lib\site-packages (3.7.1)
You are using pip version 19.0.3, however version 19.3 is available.
You should consider upgrading via the 'python -m pip install --upgrade pip' command.
```

Step 9 Enter the command **python** to enter interactive mode, notice the version of python that you installed and that the prompt is now a >>>.

```
C:\Users\student>python
Python 3.6.7 (v3.6.7:6ec5cf24b7, Oct 20 2018, 13:35:33) [MSC v.1900 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> _
```

Step 10 Lets make sure our modules can be loaded correctly by entering the following 3 commands following each with a Carriage Return. Ensure that no errors are generated.

- **import requests**
- **import urllib3**
- **import configparser**

 © 2020 NterOne Corporation

```
>>> import requests
>>> import urllib3
>>> import configparser
>>>
```

Step 11 Enter the following in interactive mode, Python program print command **print ("Hello World!")** and press **<Enter>** and notice "Hello World!" is output.

```
>>>print ("Hello World!")
Hello World!
>>>
```

Note At any point in time, to exit the Python interactive mode prompt, press **<CTRL>** + **d** or type **exit()** and then **<Enter>**.

Step 12 Enter From the command prompt session, configure two Python variables "a" and "b". Assign the value to each as follows:

- Variable a value: "SDWAN"
- Variable b value: 1234

Note A single variable is used to represent a piece of data or a set of data.

From the Terminal session in the interactive Python mode on the Admin PC enter the **a="SDWAN"** command and press **<Enter>**. Then, print the value of variable **a** using the **print(a)** command. Press **<Enter>** again to see the result.

Configure using the value of variable **b** by entering the command **b=1234 <Enter>**. Then print the value of variable b using the command **print(b) <Enter>**. The output should look similar to this:

```
>>> a="SDWAN"
>>> print(a) SDWAN
>>> b=1234
>>> print(b) 1234
```

Note This output shows that a variable can be assigned to a text value (delimited by single or double quotes) or a numerical value (with no quotes). In the examples, you use the print statement to show the contents of a variable. In the Python interpreter, you can examine the contents of a variable by simply typing the variable name.

Step 13 From the command prompt session, configure a new Python list variable **c.** Then, define the following list items values for index values 0, 1, and 2:

- c[0] = "Cisco"
- c[1] = 'SDWAN'
- c[2] =4321

In the command prompt session, with the Python interpreter in the interactive mode, enter the **c=["Cisco", 'SDWAN',4321] <Enter>**. This creates a new list with **Cisco** in index 0 first, then **SDWAN** in index 1, and then **4321** in index 2.

In order to output each value of the list **c**, enter the **c[0] <Enter>**, then **c[1] <Enter>**, and finally **c[2] <Enter>**. You should see the following result:

```
>>> c=["Cisco",'SDWAN',4321] <Enter>
>>> c[0] <Enter>
'Cisco'

>>> c[1] <Enter>

'SDWAN'

>>> c[2] <Enter> 4321
```

Step 14 From the command prompt session, configure Python **for** loop statement that will print the 3 indexes that you defined in the list called c.

Enter the **for item in c: <Enter> <Tab>print(item) <Enter>** command and then **<Enter>** again after the three dots in interpretive mode as follows to output the content of the **c** list. Your output should be similar to this:

```
>>> for item in c:<Enter>
…     <tab>print(item)<Enter>

...<Enter> Cisco SDWAN 4321

>>>
```

Step 15 From the command prompt session, configure and use a Python for loop statement to print a range of values from 1 to 3.

Note If you are familiar with for loop statements from another language, they may look a little different in Python. You probably remember that loops are used for iterating through a series of numbers. We can do this in Python as well, but we need to use a special syntax.

Python **for** loop operator iterates some integer variable in increasing or decreasing order. Such a sequence of integer can be created using the function **range(min_value, max_value)**.

The Python **range(min_value, max_value)** function generates a list of numbers, which is generally used to iterate over with **for** loops. Function **range(min_value, max_value)** generates a sequence with numbers **min_value**, **min_value + 1**, **…**, **max_value – 1**. The last number is not included.

 © 2020 NterOne Corporation

Enter the following **for** loop statement **for n in range(1,4):<Enter><tab>print (n)<Enter>** and then again **<Enter>** after the 3 dots in the interactive mode and notice that the output of this statement are printed integer values 1, 2, and 3.

The result may surprise you. Python stops before it gets to 4. This is normal, and you have to take this into account when using the for loop statement range. In most cases this means the range max value should be set to n+1 of you want to iterate through the loop n times.

Your output should be similar to this:

```
>>> for n in range(1,4):<Enter>
...     <tab>print(n)<Enter>
...<Enter>
1
2
3
```

Step 16 From the Admin PC cmd session, you will create and store data in a dictionary data structure called switch, which is similar to a list data structure. This dictionary will store the following paramters:

- vEdge: **Branch-vE1**
- IP address: **10.82.83.111**
- Username: **admin**
- Password: **admin**

Note Unlike a list, however, you can retrieve information from dictionaries using keys. Instead of a number we can access specific elements of a dictionary with key names. We still enclose the key in brackets, as we did with lists.

Enter the following in the Python interpreter interactive mode:

```
>>> vEdge={"name":"branch-vE1","ip":"10.82.83.111","username":"admin","password":"admin"}
<Enter>
>>> vEdge["name"] <Enter>
'branch-vE1'
>>> vEdge["password"] <Enter>
'admin'
```

Note If you are having trouble with the dictionary, be sure you used curly brackets to define it, that you enclosed each key and value with quotes, and that you put commas between each value. Also, note that while we define the dictionary with curly brackets, we access each element with square brackets.

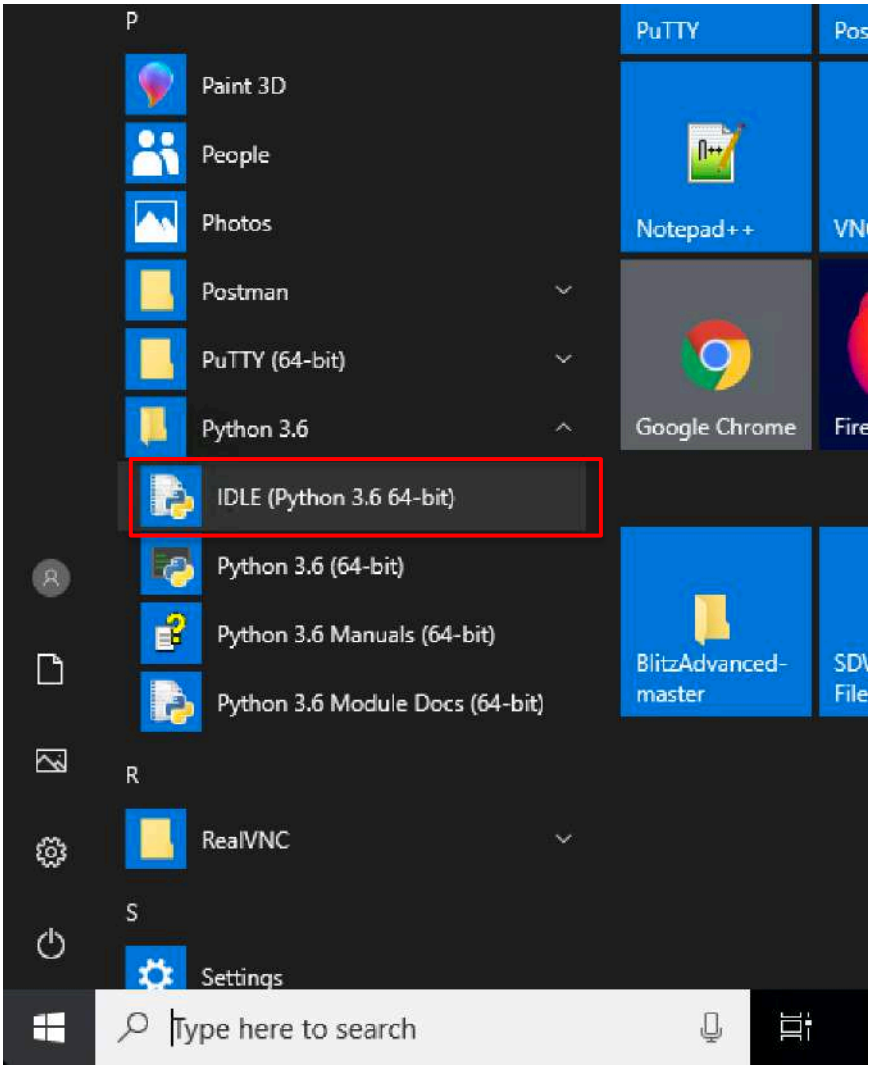
Task 2: Examine and Run a Sample Python Script and View the Output

In this task you will open and run sample python script to demonstrate some of what is possible thru the API.

Activity Procedure

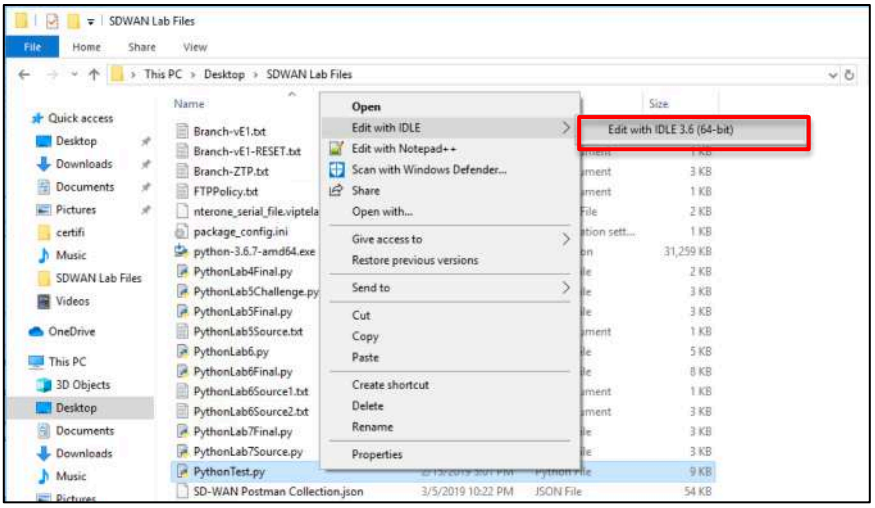
Complete these steps:

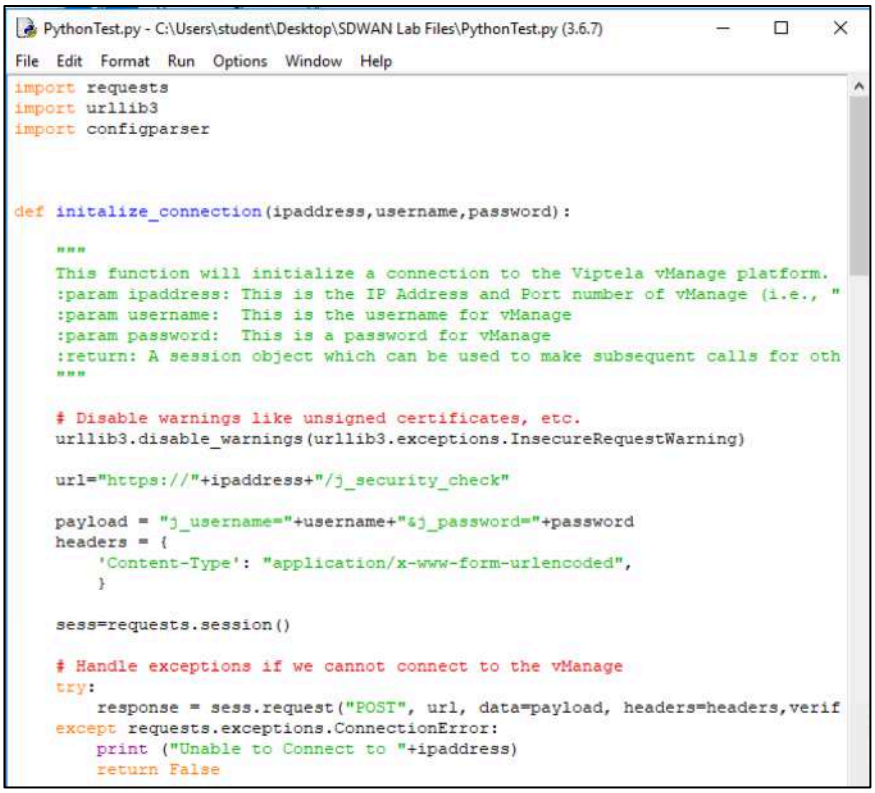
Step 1 From the remote desktop of your admin PC navigate to the start button and go to **Start > All Programs >** Python 3.6 > IDLE (Python 3.6 64-bit).



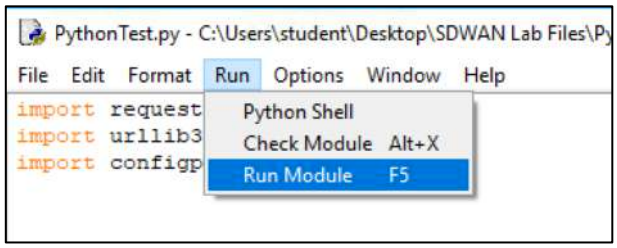
Step 2 From the IDLE application go to **File > Open** and navigate and select the script called **PythonTest.py** located in the **SDWAN Lab Files** folder on your desktop.

 © 2020 NterOne Corporation



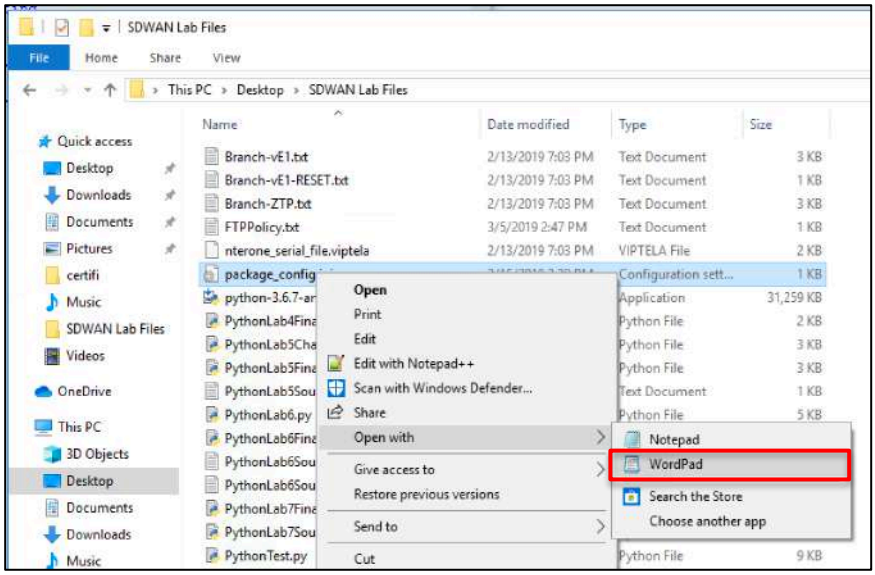


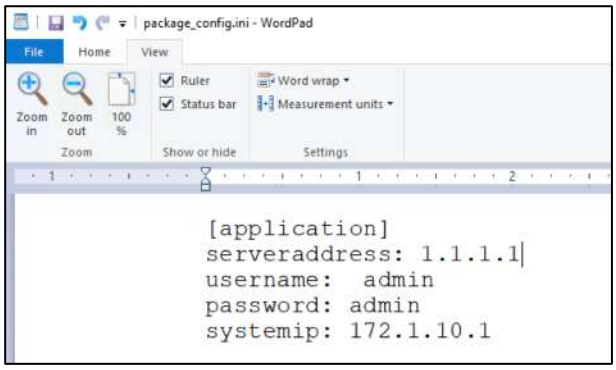
Step 3 From the IDLE application select **Run > Run Module** to execute the python script. Notice that the script is trying to connect to the wrong device.



```
Python 3.6.7 Shell
File  Edit  Shell  Debug  Options  Window  Help
Python 3.6.7 (v3.6.7:6ec5cf24b7, Oct 20 2018, 13:35:33) [MSC v.1900 64 bit (AMD6
4)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
====== RESTART: C:\Users\student\Desktop\SDWAN Lab Files\PythonTest.py ======
Viptela vManage Engine Starting...

Viptela Configuration:
vManage Server Address: 1.4.28.28
vManage Username: admin
Unable to Connect to 1.4.28.28
>>>
```

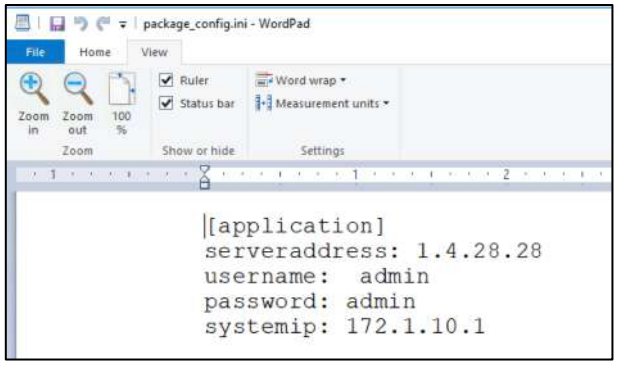
Step 4 Open up in Wordpad (not notepad) the file named **package_config.**ini, also located in the **SDWAN Lab Files** folder on your Desktop and view the contents.



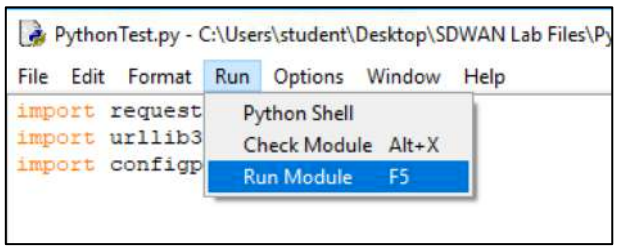


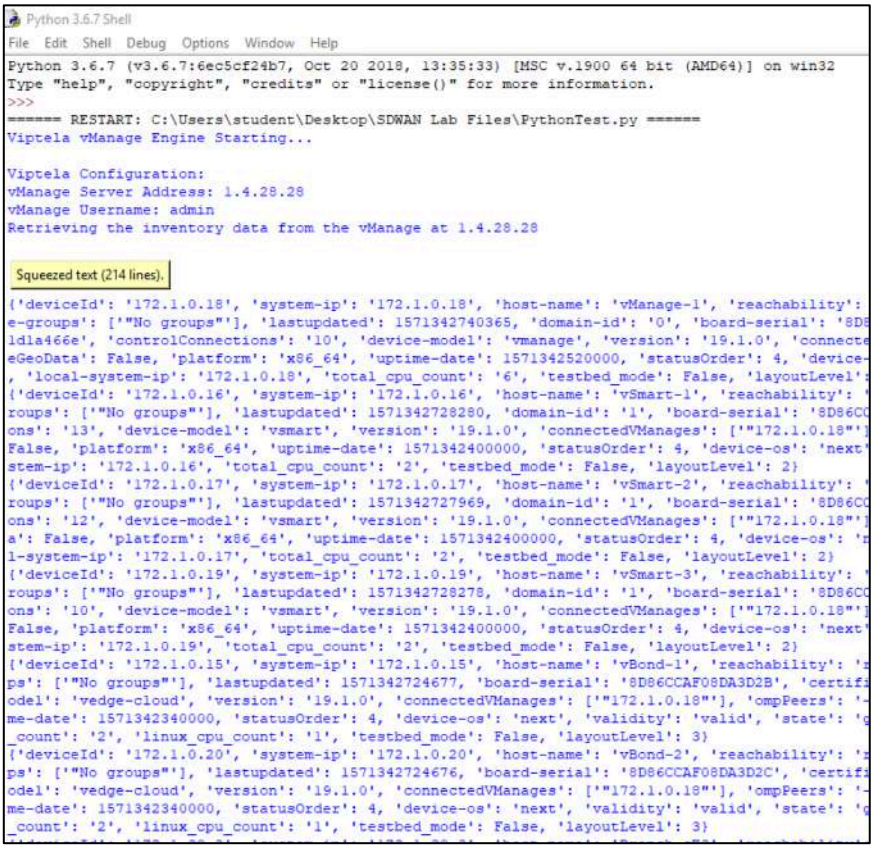
Step 5 Fix the package_config.ini file by changing the IP address from 1.1.1.1 to 1.4.28.28. Save the file when done.

 © 2020 NterOne Corporation



Step 6 Back in the PythonTest.py IDLE window select **Run > Run Module** to execute the python script. Notice that this time the script connects and starts displaying output. The inventory of vManage is displayed with this script, followed by the statistics of the 172.16.10.1(Branch-vE1) and then finally the tunnel statistics for every tunnel on the Branch-vE1 router.





Lab 15: Deploy Multitenant vManage

In this Lab, you will provision and deploy Multitenant vManage.

Activity Procedure

The following devices and values will be used in this Lab. **(The saved session shortcuts in Putty may be incorrect!!!). I would suggest fixing the ip addresses in putty first.**

Also, the \system32\drivers\etc\hosts file in the admin pc will require an additional line added:

10.82.83.101 nterone1.demo.local

You will need to be logged into the admin pc with Administrator privileges.

Device Name	IP Address or URL	Username	Password
vManage-MT	172.1.100.101 – SSH 10.82.83.101	admin	admin
vBond-MT	172.1.100.102 – SSH 10.82.83.102	admin	admin
vSmart-MT	172.1.100.103 – SSH 10.82.83.103	admin	admin

Command	Value
host-name	vManage-MT
system-ip	172.1.1.101
site-id	1500
sp-organization-name	demo.local
organization-name	demo.local
vbond	172.1.100.102
int eth1 (VPN 0)	172.1.100.102 /24
int eth0 (SSH)	10.82.83.101 /20

 © 2020 NterOne Corporation

Command	Value
host-name	vBond-MT
system-ip	172.1.1.102
sp-organization-name	demo.local
vbond	172.1.100.102 local
int ge0/0 (VPN 0)	172.1.100.102/24
int eth0 (SSH)	10.82.83.102 /20

Command	Value
host-name	vSmart-MT
system-ip	172.1.1.103
site-id	2001
controller-group	10
sp-organization-name	demo.local
organization-name	“NterOne1”
vbond	172.1.100.102
int eth1 (VPN 0)	172.1.100.103 /24
int eth0 (SSH)	10.82.83.103 /20

Task 1: Deploy Multitenant vManage

In this task you will provision a basic configuration for vManage and vBond, and you will enable Multitenant Mode on vManage.

Activity Procedure

Complete these steps:

Using **SSH** log in to your **Pod’s vManage-MT** with credentials from the table. (The saved session shortcuts in Putty may be incorrect!!!)

Step 1 Verify the configuration by issuing the command **“show run”**.

```
vManage# show run
system
 host-name vManage-MT
 admin-tech-on-failure
 aaa
  auth-order local radius tacacs
  usergroup basic
   task system read write
   task interface read write
  !
  <output omitted>
              !
          logging
         disk
            enable
       !
              !
```

```
  !
vpn 0
!
vpn 512
       interface eth0
ip address 10.82.83.101/20
no shutdown
       !
  !
vManage#
```

Step 2 Enter the following commands along with their appropriate values.

```
vManage# config
Entering configuration mode terminal
vManage(config)# system
vManage(config-system)# host-name vManage-MT
vManage(config-system)# system-ip 172.1.1.101
vManage(config-system)# site-id 1500
vManage(config-system)# sp-organization-name demo.local
vManage(config-system)# organization-name demo.local
vManage(config-system)# vbond 172.1.100.102
vManage(config-system)# exit
vManage(config)# vpn 0
vManage(config-vpn-0)# ip route 0.0.0.0/0 172.1.100.254
vManage(config-vpn-0)# interface eth1
vManage(config-interface-eth1)# ip address 172.1.100.101/24
vManage(config-interface-eth1)# no shut
vManage(config-interface-eth1)# tunnel-interface
vManage(config-tunnel-interface)# exit
vManage(config-interface-eth1)# exit
vManage(config-vpn-0)# exit
vManage(config)#
```

Step 3 **Verify that your Pod's vManage-MT configuration** is correct. Modify any values as necessary and then **commit**.

```
vManage(config)# do show configuration
system
 system-ip          172.1.1.101
 site-id            1500
 sp-organization-name  demo.local
 organization-name     demo.local
 vbond 172.1.100.102
!
```

 © 2020 NterOne Corporation

```
vpn 0
 interface eth1
  ip address 172.1.100.101/24
  tunnel-interface
   allow-service dhcp
   allow-service dns
   allow-service icmp
   no allow-service sshd
   no allow-service netconf
   no allow-service ntp
   no allow-service stun
   allow-service https
   !
   no shutdown
   !
 ip route 0.0.0.0/0 172.1.100.254
 !
```

Step 4 Now commit the configuration

```
vManage(config)# commit and-quit

Commit complete.

vManage-MT#
```

Using **SSH,** log in to your **Pod's vBond-MT** with credentials from the table. (The saved session shortcuts in Putty may be incorrect!!!)

Step 5 Verify the configuration by issuing the command **"show run"**. vBond devices start out as a vEdge router.

```
vedge# show run
      system
       host-name                vedge
       admin-tech-on-failure
       no route-consistency-check
       vbond ztp.viptela.com
       aaa
        auth-order local radius tacacs
        usergroup basic
         task system read write
         task interface read write
```

```
  !
  usergroup netadmin
  !
  usergroup operator
   task system read
   task interface read
   task policy read
   task routing read
   task security read
  !
  usergroup tenantadmin
  !
  user admin
   password
$6$irQQexn/lgd3Q8Rg$AD9mBKvQawlb0W/uypKgwk9x9UAT7eypR/n
/7OheIYlBOhRq16C6ugA8HpJbZHmk51m2iitftsn5lYCI7YpYV.
  !
 !
 logging
  disk
   enable
  !
 !
!
omp
 no shutdown
 graceful-restart
 advertise connected
 advertise static
!
security
 ipsec
  authentication-type ah-sha1-hmac sha1-hmac
 !
!
vpn 0
 interface ge0/0
  ip dhcp-client
  ipv6 dhcp-client
  tunnel-interface
   encapsulation ipsec
   no allow-service bgp
   allow-service dhcp
   allow-service dns
   allow-service icmp
   no allow-service sshd
```

 © 2020 NterOne Corporation

```
    no allow-service netconf
    no allow-service ntp
    no allow-service ospf
    no allow-service stun
    allow-service https
   !
   no shutdown
  !
 !
 vpn 512
  interface eth0
   ip address 10.82.83.102/20
   ipv6 dhcp-client
   no shutdown
  !
 !
 vedge#
```

Step 6 Enter the following commands along with their appropriate values.

```
vedge# config
Entering configuration mode terminal
vedge(config)# system
vedge(config-system)# host-name vBond-MT
vedge(config-system)# system-ip 172.1.1.102
vedge(config-system)# sp-organization-name demo.local
vedge(config-system)# organization-name demo.local
vedge(config-system)# vbond 172.1.100.102 local
vedge(config-system)# exit
vedge(config)# vpn 0
vedge(config-vpn-0)# ip route 0.0.0.0/0 172.1.100.254
vedge(config-vpn-0)# interface ge0/0
vedge(config-interface-ge0/0)# ip address 172.1.100.102/24
vedge(config-interface-ge0/0)# no shut
vedge(config-interface-ge0/0)# no tunnel-interface
vedge(config-interface-ge0/0)# exit
```

vedge(config-vpn-0)# **exit**

Step 7 Verify that your Pod's vBond configuration is correct. Modify any values as necessary and then commit.

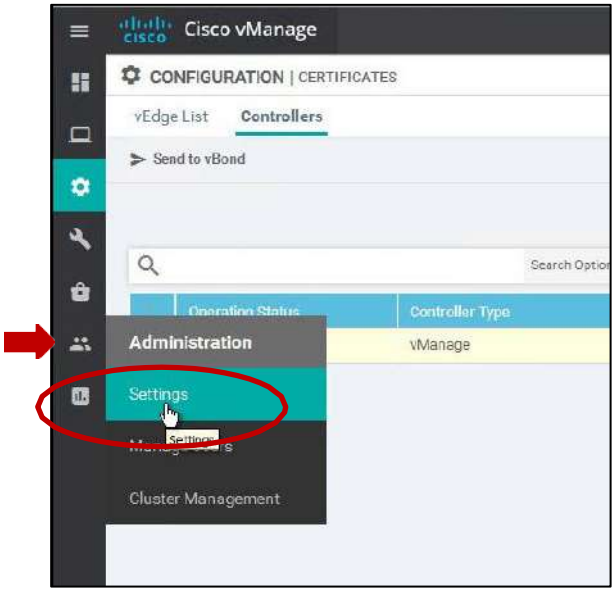
```
vedge(config)# show configuration
system
 host-name             vBond-MT
 system-ip             172.1.1.102
 sp-organization-name demo.local
 organization-name     demo.local
 vbond 172.1.100.102 local
!
vpn 0
 interface ge0/0
  ip address 172.1.100.102/24
 !
 ip route 0.0.0.0/0 172.1.100.254
!
vedge(config)# commit and-quit
Commit complete.
vBond-MT#
```

Step 8 Open a browser window and **login to vManage (https://**10.82.83.101:8443**)**

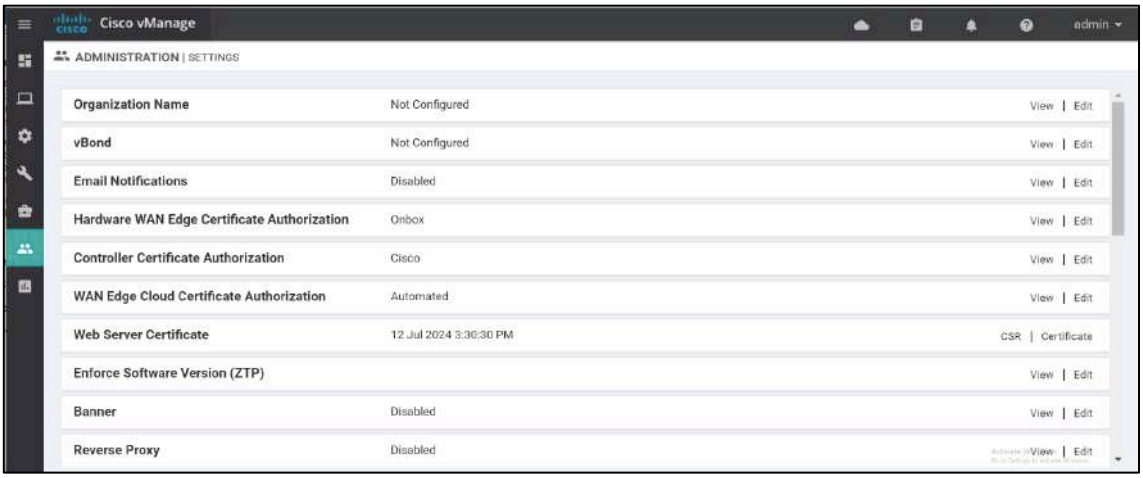
Step 9 Once logged in you'll be redirected to the Dashboard.

Step 10 Go to **Administration > Settings.**

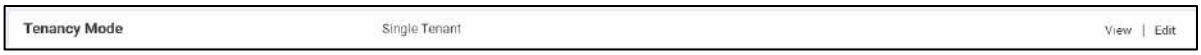
 © 2020 NterOne Corporation



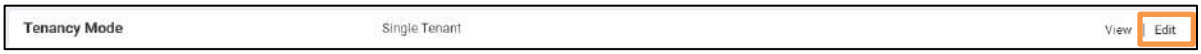
Step 11 You'll be redirected to the **System Settings** window.



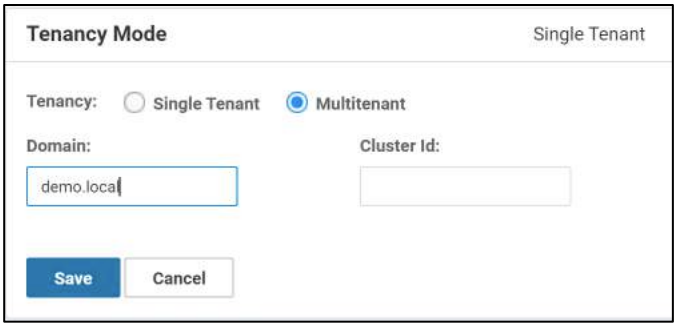
Step 12 Next we will scroll down to the section called **Tenancy Mode.** Please observe that the current mode is set to **Single Tenant**.



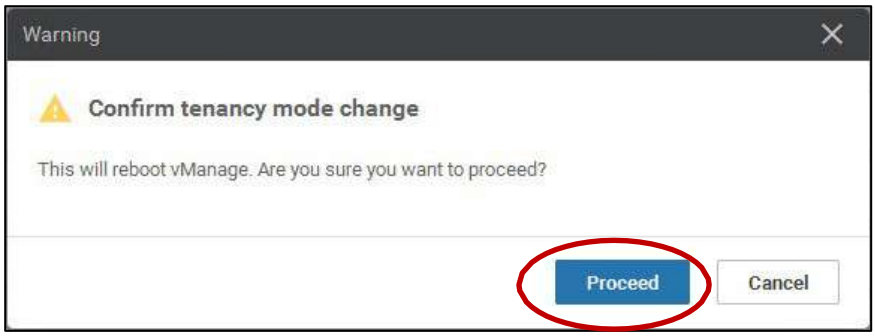
Step 13 Select **Edit** to edit the tenancy mode.



Step 14 Select **Multitenant** and for Domain: enter **demo.local.** (**Case sensitive**)) Click **Save**.



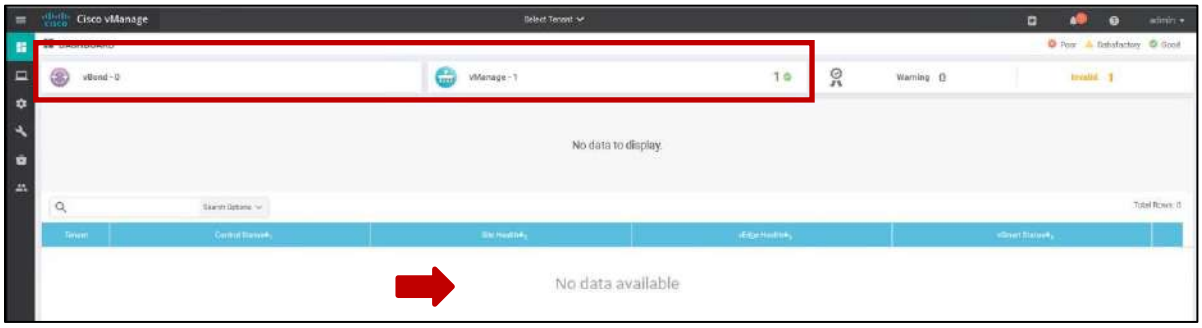
Step 15 Select **Proceed** to change the tenancy mode. Once you proceed this cannot be reversed.



Step 16 **vManage will now reboot to apply the changes. Allow a few minutes to pass before logging back in.**

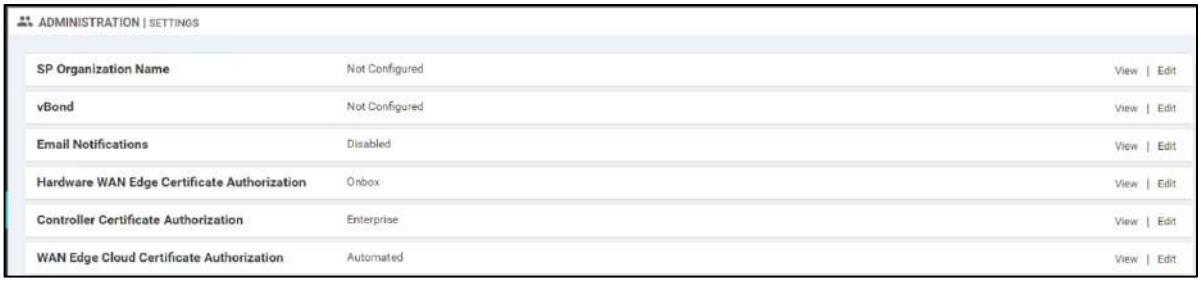
Step 17 **Log back into vManage**. The Dashboard is formatted differently in Multitenant Mode. The **Device quick view** area at the top of the window contains only vManage and vBond devices, which are managed by the Service Provider.

The **Tenant** area (arrow) is located in the middle of the window. Once tenants are added to vManage, they will be displayed here (Provider view).

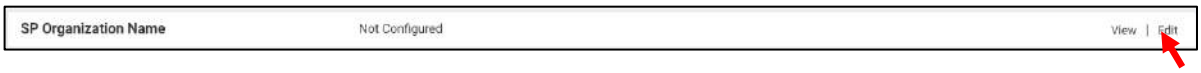


Step 18 Navigate to the **Administration > Setting** menu again. You will see that this screen has changed somewhat as well.

 © 2020 NterOne Corporation

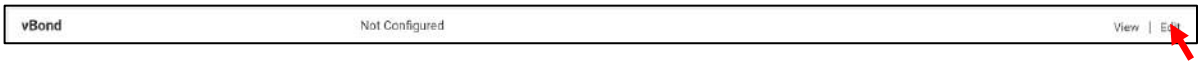


Step 19 We will now provide information to support the Multi-Tenant deployment. This will include the assignment of the SP Organizational Name, identifying the vBond. We will begin by selecting **Edit under the SP Organization Name panel.**



Now enter the Organizational Name of ***demo.local*** in twice and click **Save**.

Once that is complete select **edit for the vBond**



The vBond is located at ***172.1.100.102***, enter this number in the **vBond DNS/IP Address** field. Leave the Port at the default of ***12346***. Once completed click **Save**.

Task 2: Install vManage Certificates

You will install a certificate on vManage to be used for establishing control-plane connectivity.

Activity Procedure

Complete these steps:

Step 1 First, we need to generate and install a **Root Certificate on vManage**. We will do this manually and we will use the organization name **demo.local** in the Lab. This root certificate will allow vManage to trust any certificates signed by the CA.

Step 2 Using **SSH** log in to your Pod's vManage**.**

Step 3 Once you have logged into your vManange console type: **vshell**

```
vManage-MT# vshell

vManage-MT:~$
```

You are now in the Linux bash shell of the vManage and as such you can execute Linux commands. OPENSSL is a Linux application that we will use to create our Root Certificate first we will need a Root CA key. We will do this by typing the following command:

```
vManage-MT:~$ openssl genrsa -out ROOTCA.key 2048

Generating RSA private key, 2048 bit long modulus

............

.

e is 65537 (0x10001)

vManage-MT:~$
```

Now we will use this key to generate the ROOT CA certificate. We will generate a certificate with 5 years of validity. Please use the exact information illustrated below or your Certificate will not work.

```
vManage-MT:~$ openssl req -x509 -new -nodes -key ROOTCA.key -
sha256 -days 2000 -out ROOTCA.pem

You are about to be asked to enter information that will be
incorporated into your certificate request.

What you are about to enter is what is called a Distinguished
Name or a DN.

There are quite a few fields but you can leave some blank

For some fields there will be a default value,

If you enter '.', the field will be left blank.

-----

Country Name (2 letter code) [AU]:US

State or Province Name (full name) [Some-State]:North Carolina
```

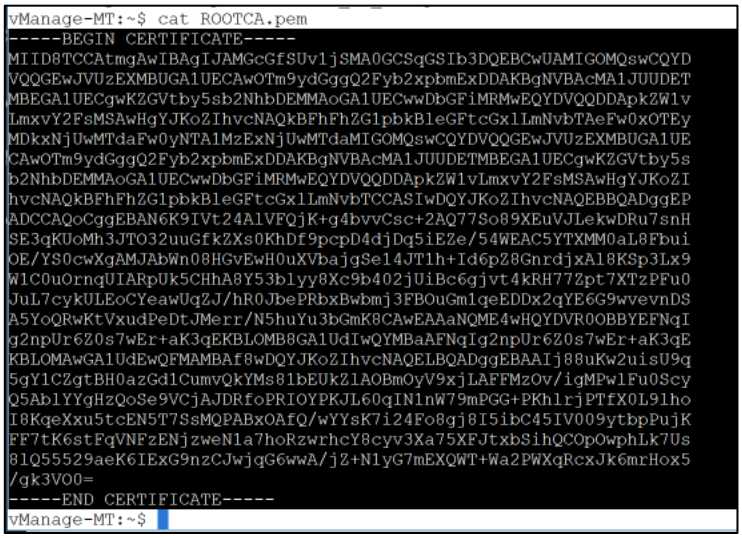
 © 2020 NterOne Corporation

```
Locality Name (eg, city) []:RTP

Organization Name (eg, company) [Internet Widgits Pty
Ltd]:demo.local

Organizational Unit Name (eg, section) []:lab

Common Name (e.g. server FQDN or YOUR name) []:demo.local

Email Address []:admin@example.com

vManage-MT:~$
```

This has created a new file in the current folder called ROOTCA.pem. We will confirm this via the ls command:

```
vManage-MT:~$ ls

ROOTCA.key   ROOTCA.pem   archive_id_rsa.pub

vManage-MT:~$
```

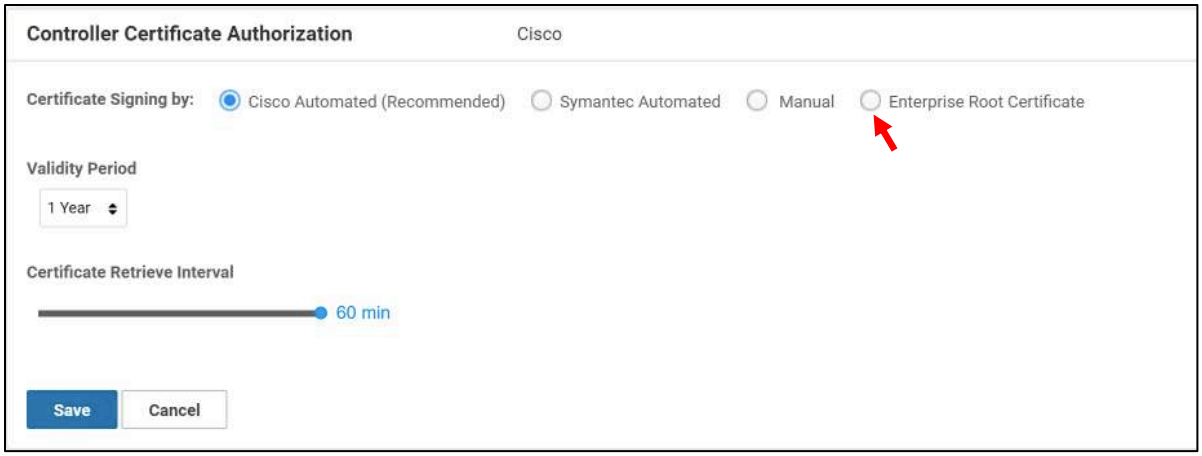
We will now look at the key we just created. Actually, we will copy and paste the key into our vManage Administrative Settings: You can use **#vi ROOTCA.pem**



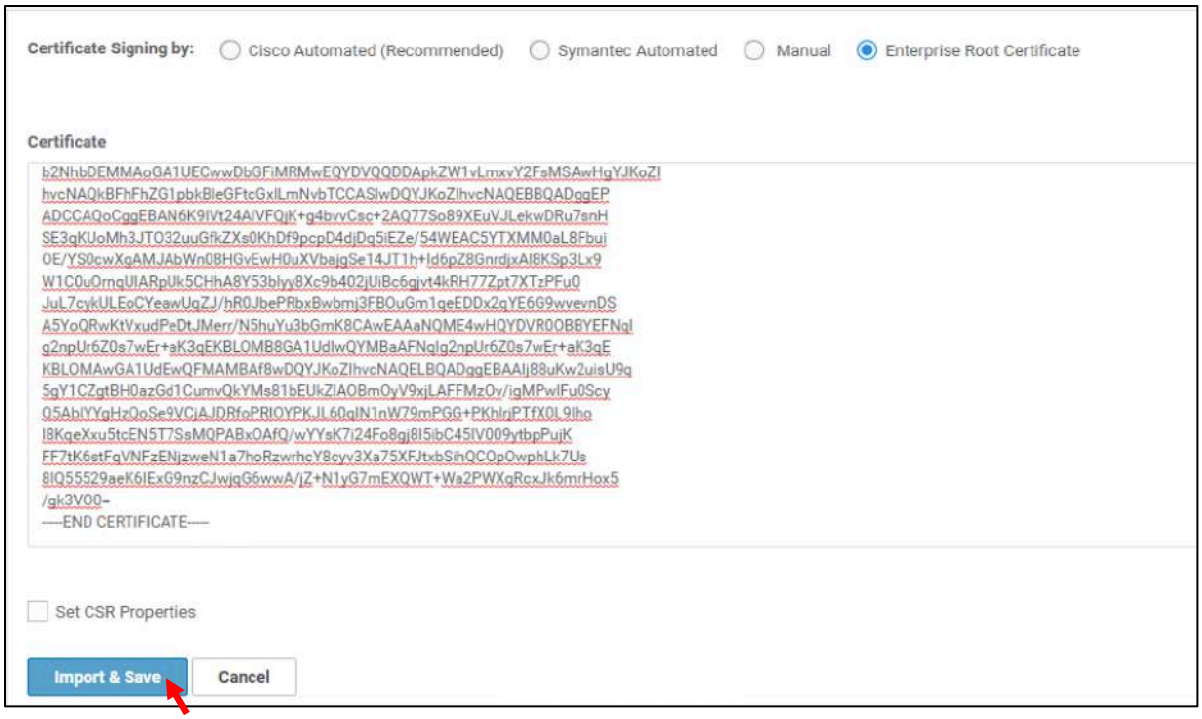
```
vManage-MT:~$ cat ROOTCA.pem
-----BEGIN CERTIFICATE-----
MIID8TCCAtmgAwIBAgIJAMGcGfSUv1jSMA0GCSqGSIb3DQEBCwUAMIGOMQswCQYD
VQQGEwJVUzEXMBUGA1UECAwOTm9ydGggQ2Fyb2xpbmExDDAKBgNVBAcMA1JUUDET
MBEGA1UECgwKZGVtby5sb2NhbDEMMAoGA1UECwwDbGFiMRMwEQYDVQQDDApkZW1v
LmxvY2FsMSAwHgYJKoZIhvcNAQkBFhFhZG1pbkBleGFtcGxlLmNvbTAeFw0xOTEy
MDkxNjUwMTdaFw0yNTA1MzExNjUwMTdaMIGOMQswCQYDVQQGEwJVUzEXMBUGA1UE
CAwOTm9ydGggQ2Fyb2xpbmExDDAKBgNVBAcMA1JUUDETMBEGA1UECgwKZGVtby5s
b2NhbDEMMAoGA1UECwwDbGFiMRMwEQYDVQQDDApkZW1vLmxvY2FsMSAwHgYJKoZI
hvcNAQkBFhFhZG1pbkBleGFtcGxlLmNvbTCCASIwDQYJKoZIhvcNAQEBBQADggEP
ADCCAQoCggEBAN6K9IVt24AlVFQjK+g4bvvCsc+2AQ77So89XEuVJLekwDRu7snH
SE3qKUoMh3JTO32uuGfkZXs0KhDf9pcpD4djDq5iEZe/54WEAC5YTXMM0aL8Fbui
OE/YS0cwXgAMJAbWn08HGvEwH0uXVbajgSe14JT1h+Id6pZ8GnrdjxAl8KSp3Lx9
W1C0uOrnqUIARpUk5CHhA8Y53blyy8Xc9b402jUiBc6gjvt4kRH77Zpt7XTzPFu0
JuL7cykULEoCYeawUqZJ/hR0JbePRbxBwbmj3FBOuGm1qeEDDx2qYE6G9wvevnDS
A5YoQRwKtVxudPeDtJMerr/N5huYu3bGmK8CAwEAAaNQME4wHQYDVR0OBBYEFNqI
g2npUr6Z0s7wEr+aK3qEKBLOMB8GA1UdIwQYMBaAFNqIg2npUr6Z0s7wEr+aK3qE
KBLOMAwGA1UdEwQFMAMBAf8wDQYJKoZIhvcNAQELBQADggEBAAIj88uKw2uisU9q
5gY1CZgtBH0azGd1CumvQkYMs81bEUkZ1AOBmOyV9xjLAFFMzOv/igMPwlFu0Scy
Q5AblYYgHzQoSe9VCjAJDRfoPRIOYPKJL60qIN1nW79mPGG+PKhlrjPTfX0L9lho
I8KqeXxu5tcEN5T7SsMQPABxOAfQ/wYYsK7i24Fo8gj8I5ibC45IV009ytbpPujK
FF7tK6stFqVNFzENjzweN1a7hoRzwrhcY8cyv3Xa75XFJtxbSihQCOpOwphLk7Us
81Q55529aeK6IExG9nzCJwjqG6wwA/jZ+N1yG7mEXQWT+Wa2PWXqRcxJk6mrHox5
/gk3VO0=
-----END CERTIFICATE-----
vManage-MT:~$
```

Step 4 Return to the vManage-MT console under the **Administration > Settings** menu. We will now navigate to the **Controller Certificate Authorization** section and select Edit.

Now we will now change the default Controller Certification Authorization from **Cisco Automated to Enterprise Root Certificate:**



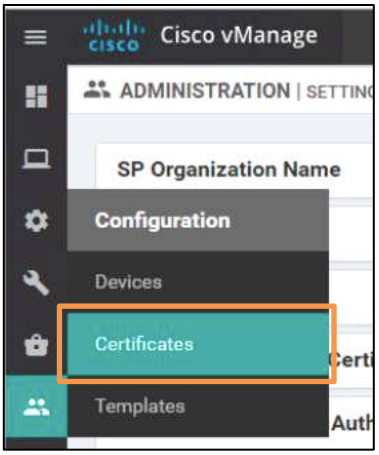
Once we do this and we **press Proceed into** resulting popup, we will be presented with the option to paste the contents of our ROOTCA.pem file into the Administration Settings page.



Leave the **"Set CSR Properties"** unchecked and press **Import & Save**.

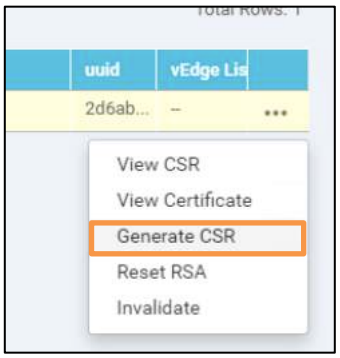
Step 5 We will now need to create a CSR for the vManage. Navigate to **Configuration > Certificates**.

 © 2020 NterOne Corporation

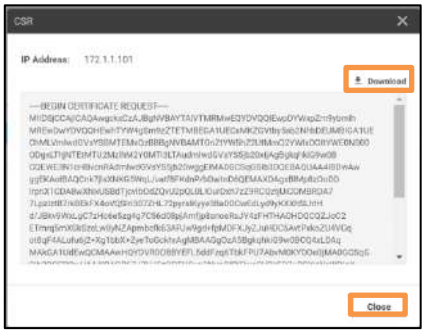


You will see the following (observe that there is no Certificate Installed at this time:

Click the …, and select Generate CSR from the resulting menu:



This will result in a popup window that looks like this:



We will download this file and then click Close

Step 6 Now we will ssh back to the vManage-MT (10.82.83.101). Once there we will need to look at some of the subtle changes that have taken place in the directory where we created the **ROOTCA.key** and **ROOTCA.pem** file. Now that we have created a CSR inside the

vManage NMS User Interface we will find a new file called **vmanage_csr**. This is the file we will use the **ROOTCA.pem** to sign in order to create a certificate for the vManage-MT. We can see this new file using the **ls** command:

```
vManage-MT# vshell

vManage-MT:~$ ls -l

total 16

-rw-r--r-- 1 admin admin 1675 Dec 10 12:37 ROOTCA.key

-rw-r--r-- 1 admin admin 1428 Dec 10 12:38 ROOTCA.pem

-rw-r--r-- 1 admin admin  394 Dec  9 17:28 archive_id_rsa.pub

-rw-r--r-- 1 root  root  1216 Dec 10 12:44 vmanage_csr

vManage-MT:~$
```

Now the time has come to sign the csr:

```
vManage-MT:~$ openssl x509 -req -in vmanage_csr -CA ROOTCA.pem
-CAkey ROOTCA.key -CAcreateserial -out vmanage.crt -days 2000
-sha256
```

```
Signature ok

subject=/C=US/ST=California/L=San Jose/OU=demo.local/O=Viptela
LLC/CN=vmanage-2d6ab18b-aa45-4881-8c51-15639a3f4127-
0.viptela.com/emailAddress=support@viptela.com

Getting CA Private Key
```

```
vManage-MT:~$ cat vmanage.crt
-----BEGIN CERTIFICATE-----
MIID1TCCAr0CCQCu6rrpuM906DANBgkqhkiG9w0BAQsFADCBjjELMAkGA1UEBhMC
VVMxFzAVBgNVBAgMDk5vcnRoIENhcm9saW5hMQwwCgYDVQQHDANSVFAxEzARBgNV
BAoMCmRlbW8ubG9jYWwxDDAKBgNVBAsMA2xhYjETMBEGA1UEAwwKZGVtby5sb2Nh
bDEgMB4GCSqGSIb3DQEJARYRYWRtaW5AZXhhbXBsZS5jb20wHhcNMTkxMjEwMTI1
NjIyWhcNMjUwNjAxMTI1NjIyWjCByTELMAkGA1UEBhMCVVMxEzARBgNVBAgTCkNh
bGlmb3JuaWExETAPBgNVBAcTCFNhbiBKb3NlMRMwEQYDVQQLEwpkZW1vLmxvY2Fs
MRQwEgYDVQQKEwtWaXB0ZWxhIExMQzFDMEEGA1UEAxM6dm1hbmFnZS0yZDZhYjE4
Yi1hYTQ1LTQ4ODEtOGM1MS0xNTYzOWEzZjQxMjctMC52aXB0ZWxhLmNvbTEiMCAG
CSqGSIb3DQEJARYTc3VwcG9ydEB2aXB0ZWxhLmNvbTCCASIwDQYJKoZIhvcNAQEB
BQADggEPADCCAQoCggEBAKeTuOWxc0oblaov+9q1/wUp2c/9sNqK2cPpAQwBcMCD
EGUynzM6gMMiumdfUIMDzBeEjFRIF1ONy8hs511C9TalAvQuU66sPGHvNn1EJDO2
NSII4wFEMDvsunMjO0HuuQESQVfihVBIiuLfTtkcvvanKuwgrLJ7fxrQ4LAR0vJ3
3IpdeFIuG0d38kGS/1bEuALvMdzp7nODiDsLnp3TymMCZ+Onyyeh5GwljjMUdMcA
4cNAJBkmgLYROaurmZdKRLNwvDzI1kCmZtx+TrcA8nD2B35+kwMVcnJkm4cMLkC+
0/GSh1ThUaqi3yoXgAu5+7qPb5eDVttf5nJ5OgZySGsCAwEAATANBgkqhkiG9w0B
AQsFAAOCAQEAe30SvXhmyrY8n7Ikip636NFfAYVUvQLiFORHLvDSS65Whdb2TuLy
SEzkPRwxuysRNZS7AYOURb4Wtv7x72JVnqLp+oRUWJdxPKYi82ZrNu07j7OovpO9
+/+/nC/H2I7D9dHoWcwDnlHbeHtd6EJJfFv5aKgLmcpz5xWf/bdkjJtoMuNrOWv/
JWZOgtbZhMKbU3xdme9WJEjj00KoMmesnfPwkZbOMHiYDGDRYOdsBZnZ00qazEhk
0nWl2epUKXLsBL7YqLTGO9hmzIPqbxtrRZBKEYsWw/3K5iEGrQcRLVQ+B6keD+dS
sHwzYyzv/tiUyU/wiVWwIvFx/CvlshKJgw==
-----END CERTIFICATE-----
vManage-MT:~$
```

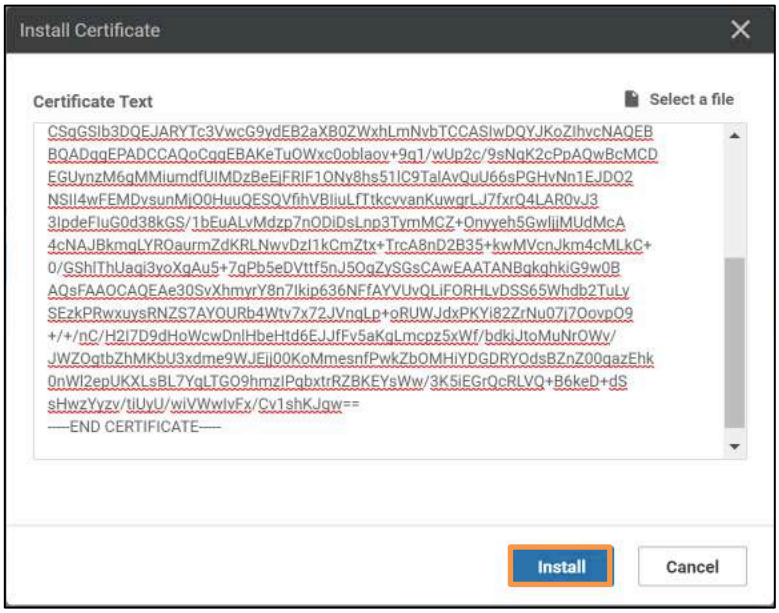
```
vManage-MT:~$
```

 © 2020 NterOne Corporation

Now we have a key we will need to copy and paste it into the vManage NMS User Interface. So, we will #**cat vmanage.crt** so we can accomplish this task.

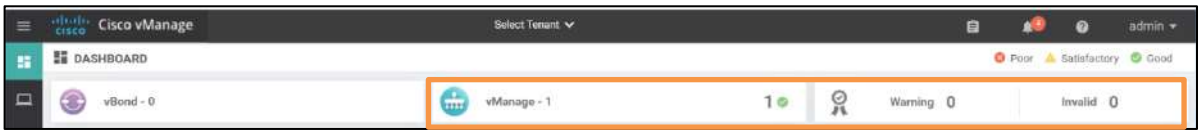
Step 7 Now we will return to the vManage dashboard **Configuration > Certificates**. You will see a button in the top right-hand of the screen named **Install Certificate** Click that.

Step 8 Paste the contents of the ***vmanage.crt*** file into that field and click **Install**.



You will be transferred to the TASK VIEW screen where after a moment or so you should see the following:

This indicates clearly that the Certificate has been installed. As a final verification we will return to the main dashboard of the vManage and we should see the following output.



If you do not see this output after a few moments notify your instructor immediately.

Step 9 End of Task 1.

Task 3: Add vBond Controller and Install Certificates

In this Task you will add the vBond controller to vManage and install a Root and Server certificate.

Activity Procedure

Complete these steps:

Step 1 First, we need to move a copy of the **ROOTCA.key** and **ROOTCA.pem** files from the vManage-MT to the vBond-MT. We will do this using the **scp** command executed from the vManage **vshell** console:

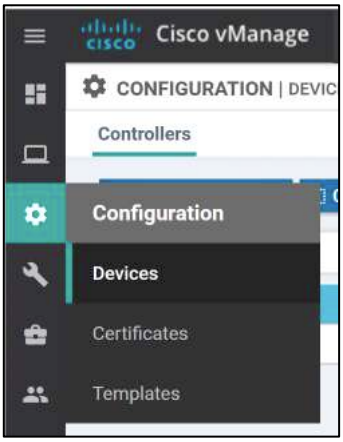
```
vManage-MT:~$ scp ROOTCA.key admin@172.1.100.102:

The authenticity of host '172.1.100.102 (172.1.100.102)' can't
be established.

ECDSA key fingerprint is
SHA256:D9jjgfU9j6xlW3FHmCtWhuSjv8l4AunjmGXivQBqaQM.

Are you sure you want to continue connecting (yes/no)? yes

Warning: Permanently added '172.1.100.102' (ECDSA) to the list
of known hosts.

viptela 19.2.099


admin@172.1.100.102's password:admin

ROOTCA.key                                    100% 1675
53.7KB/s   00:00

vManage-MT:~$ scp ROOTCA.pem admin@172.1.100.102:

viptela 19.2.099


admin@172.1.100.102's password:

ROOTCA.pem                                    100% 1428
44.7KB/s   00:00

vManage-MT:~$
```

Step 2 Now let's see if we can confirm that the files have been moved by using Putty to SSH to the console of the vBond-MT (10.82.83.102):

 © 2020 NterOne Corporation

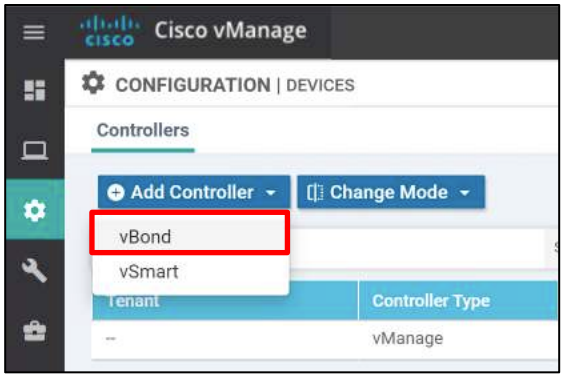
```
vBond-MT# vshell
vBond-MT:~$ ls -l
total 12
-rw-r--r-- 1 admin admin 1675 Dec 10 13:25 ROOTCA.key
-rw-r--r-- 1 admin admin 1428 Dec 10 13:25 ROOTCA.pem
-rw-r--r-- 1 admin admin  392 Dec  9 17:27 archive_id_rsa.pub
vBond-MT:~$
```

Step 3 Now we will add the vBond-MT as a controller for the current vManage configuration. This is accomplished from the vManage User Interface from the Configuration > Devices menu:

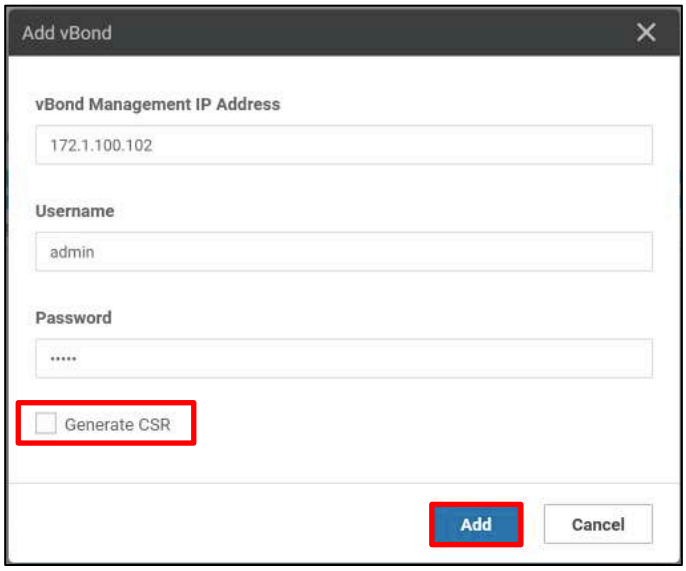


Once you click Devices you will see the following screen:

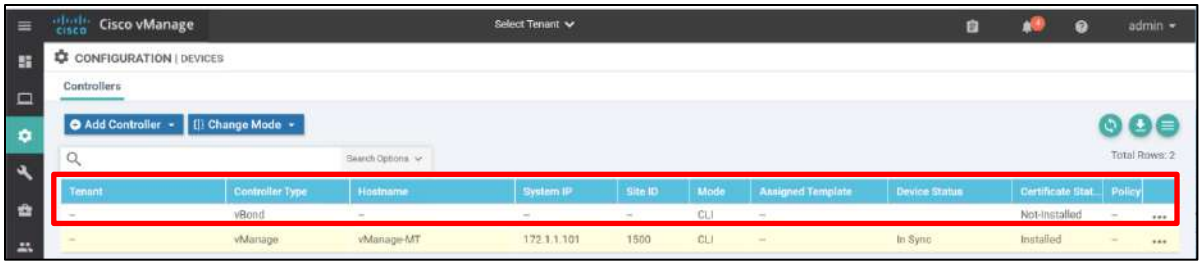
We will need to click Add Controller and select vBond from the drop down menu:



Once you add the vBond you will need to provide the necessary information for the vManage to communicate with the vBond, this will include the (VPN0) IP Address of the device, its username and password. Additionally, for the purposes of our lab we want to deselect the Generate CSR option. We will do this manually when the time comes.



Click **Add**.



Observe that the vBond-MT has been added but it does not have a Certificate installed. It is not until this device has been provided a signed certificate that we will be able to use it in our Secure Extensible Network. To verify this, we will visit the main dashboard to see what is reported there.

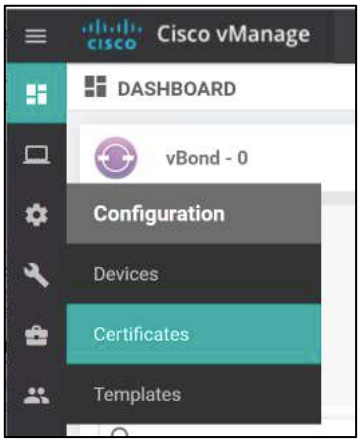
 © 2020 NterOne Corporation

Observe that we see no vBond-MT appearing in the list, and we also see that we have an invalid message. We need to take a look at it. Click the **Invalid 1** message.

Again the system is warning us that we do not have a certificate installed for the vBond.

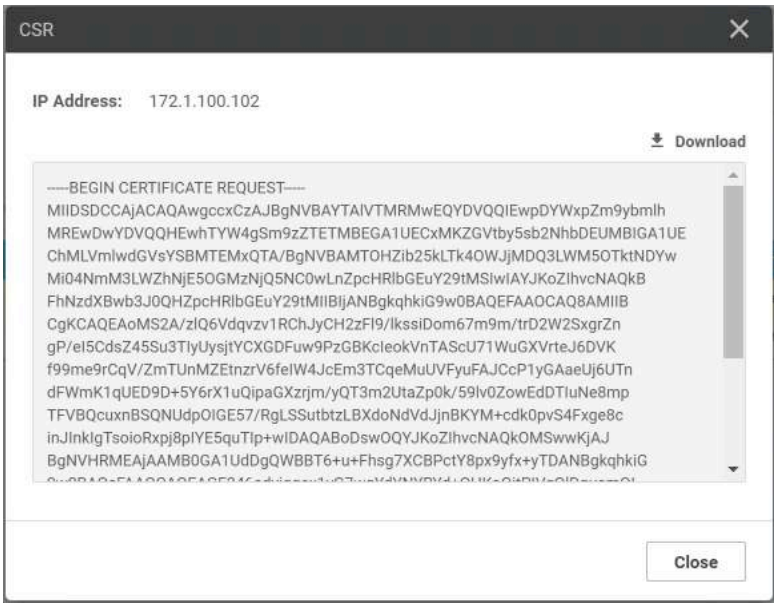
Step 4 Now we will create a CSR and sign it using the ROOTCA.PEM and ROOTCA.KEY we used to sign the certificate for the vManage.

Return to **Configuration > Certificates**:



Select the … at the end of the line for the vBond Controller:

Select the Generate CSR option in the menu.



Click Download and Close

Step 5 Use Putty to SSH to the vBond-MT (10.82.83.102) and verify that we see the vbond_csr file that we will sign using the ROOTCA.pem and ROOTCA.key files we transferred earlier. Do this with the **ls** command.

```
vBond-MT:~$ ls -l
total 48
-rw-r--r-- 1 admin admin  1675 Dec 10 13:25 ROOTCA.key
-rw-r--r-- 1 admin admin  1428 Dec 10 13:25 ROOTCA.pem
-rw-r--r-- 1 admin admin   392 Dec  9 17:27 archive_id_rsa.pub
-rwxr-xr-x 1 admin admin 24736 Dec 10 13:43 master_root.crt
-rwxr-xr-x 1 admin admin    43 Dec 10 13:43 root_ca_uuid_list
-rw-r--r-- 1 root  root   1216 Dec 10 13:58 vbond_csr
vBond-MT:~$
```

Now we will sign this certificate like we did before.

```
vBond-MT:~$ openssl x509 -req -in vbond_csr -CA ROOTCA.pem -
CAkey ROOTCA.key -CAcreateserial -out vbond.crt -days 2000 -
sha256
```

```
Signature ok
subject=/C=US/ST=California/L=San Jose/OU=demo.local/O=Viptela
LLC/CN=vbond-989bc047-c999-4602-86c7-fa6198c36494-
0.viptela.com/emailAddress=support@viptela.com
```

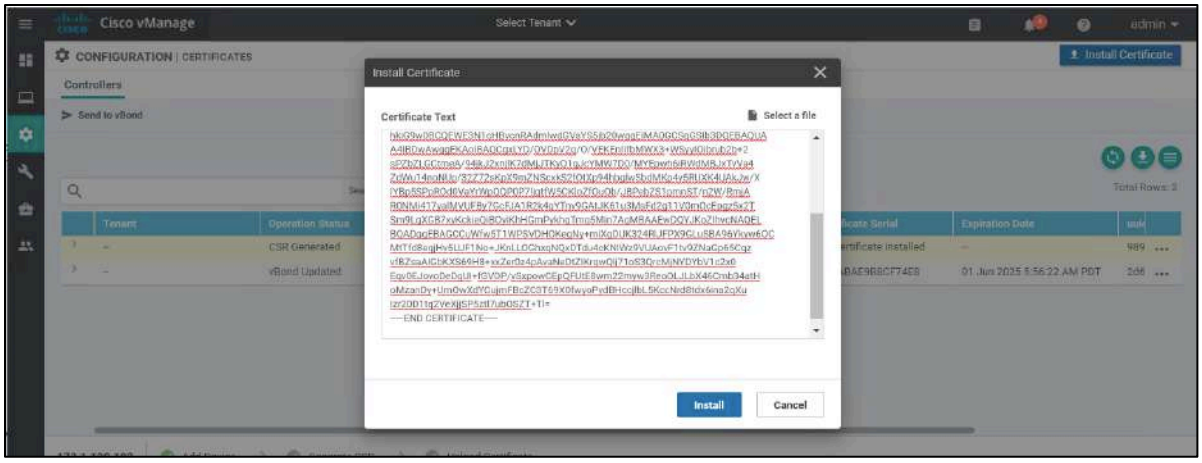
 © 2020 NterOne Corporation

```
Getting CA Private Key

vBond-MT:~$
```

Now we need to copy and paste this into the vManage NMS as we did for the vManage itself.

```
vBond-MT:~$ cat vbond.crt
-----BEGIN CERTIFICATE-----
MIID0zCCArsCCQDhd7PjRSmuJzANBgkqhkiG9w0BAQsFADCBjjELMAkGA1UEBhMC
VVMxFzAVBgNVBAgMDk5vcnRoIENhcm9saW5hMQwwCgYDVQQHDANSVFAxEzARBgNV
BAoMCmRlbW8ubG9jYWwxDDAKBgNVBAsMA2xhYjETMBEGA1UEAwwKZGVtby5sb2Nh
bDEgMB4GCSqGSIb3DQEJARYRYWRtaW5AZXhhbXBsZS5jb20wHhcNMTkxMjEwMTQw
NDMyWhcNMjUwNjAxMTQwNDMyWjCBxzELMAkGA1UEBhMCVVMxEzARBgNVBAgTCkNh
bGlmb3JuaWExETAPBgNVBAcTCFNhbiBKb3NlMRMwEQYDVQQLEwpkZW1vLmxvY2Fs
MRQwEgYDVQQKEwtWaXB0ZWxhIExMQzFBMD8GA1UEAxM4dmJvbmQtOTg5YmMwNDct
Yzk5OS00NjAyLTg2YzctZmE2MTk4YzM2NDk0LTAudmlwdGVsYS5jb20xIjAgBgkq
hkiG9w0BCQEWE3N1cHBvcnRAdmlwdGVsYS5jb20wggEiMA0GCSqGSIb3DQEBAQUA
A4IBDwAwggEKAoIBAQCgxLYD/OVDpV2q/O/VEKEnIIfbMWX3+WSyyIOibrub2b+2
sPZbZLGCtmeA/94jkJ2xnjlK7dMjJTKyO1gJcYMW7D0/MYEpwh6iRWdMBJxTvVa4
ZdWu14noNUp/32Z72sKpX9mZNScxkS2fOtXp94hbglwSbdMKp4y5RUXK4UAkJw/X
IYBp5SPpROd0VaYrWpQQP0P7ljqtfW5CKloZfOuOb/JBPebZS1pmnST/n2W/RmjA
R0NMi417yalMVUFBy7GcFJA1R2k4gYTnv9GAtJK61u3MsFd2g11V0mOcEpgz5x2T
Sm9LgXGB7xyKckieQiBOyiKhHGmPykhgTmq5Min7AgMBAAEwDQYJKoZIhvcNAQEL
BQADggEBAGCCuWfw5T1WPSVDHOKegNy+miXgDUK324RlJFPX9GLuSBA96Ykyw6OC
MtTfd8egjHv5LUF1No+JKnLLOGhxqNQxDTdu4cKNlWz9VUAovF1tv9ZNaGp65Cgz
vfBZsaAIGbKXS69H8+xxZer0z4pAvaNeDtZIKrgwQlj71oS3QrcMjNYDYbV1c2x0
Eqv0EJovoDeDgUI+fGVDP/vSxpowCEpQFUtE8wm22myw3ReoOLJLbX46Cmb34atH
oMzanDy+UmOwXdYCujmFBcZC3T69X0fwyoPydBHccjlbL5KccNrd8tdx6ina2qXu
Izr20D1tq2VeXjjSP5ztl7ubOSZT+TI=
-----END CERTIFICATE-----
vBond-MT:~$
```

Now we paste this into the **Configure > Certificate** window of the vManage and click install.



If we did everything correctly, we should see that the certificate installed in the TASK VIEW window that pops up. Be patient, but if you do not see a screen that says Success like the one illustrated below please notify your Instructor Immediately.

Now I want to explore the Main Dashboard again to see if the vBond has been added.

It has appeared and we have no messages about an invalid certificate anymore!

Step 6 Using **SSH** log in to your **Pod's vBond-MT.**

Step 7 Confirm that the certificate has been installed. Notice the Issuer (**demo.local**), and the OU which has been assigned to the **Service Provider Organization – demo.local** (**X** = Pod number)

```
vBond-MT# show certificate installed | more


Server certificate
------------------

Certificate:
    Data:
        Version: 1 (0x0)
        Serial Number:
            e1:77:b3:e3:45:29:ae:27
```

 © 2020 NterOne Corporation

```
    Signature Algorithm: sha256WithRSAEncryption
        Issuer: C=US, ST=North Carolina, L=RTP, O=demo.local, OU=lab,
CN=demo.local/emailAddress=admin@example.com
        Validity
            Not Before: Dec 10 14:04:32 2019 GMT
            Not After : Jun  1 14:04:32 2025 GMT
        Subject: C=US, ST=California, L=San Jose, OU=demo.local, O=Viptela LLC,
CN=vbond-989bc047-c999-4602-86c7-fa6198c36494-
0.viptela.com/emailAddress=support@viptela.com
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (2048 bit)
                Modulus:
                    00:a0:c4:b6:03:fc:e5:43:a5:5d:aa:fc:ef:d5:10:
                    a1:27:20:87:db:31:65:f7:f9:64:b2:c8:83:a2:6e:
                    bb:9b:d9:bf:b6:b0:f6:5b:64:b1:82:b6:67:80:ff:
                    de:23:90:9d:b1:9e:39:4a:ed:d3:23:25:32:b2:3b:
                    58:09:71:83:16:ec:3d:3f:31:81:29:c2:1e:a2:45:
<output omitted>
```

Step 8 Issue "**show orchestrator**" commands to verify connectivity to vManage.

The **System IP** is assigned by vManage.

The **SP Organization** should match the previously provisioned value **demo.local.**

The **"show orchestrator valid-vsmarts"** command should reference the vManage certificate serial no.

(If it does not display horizontally add "**| tab**" to the end of the command)

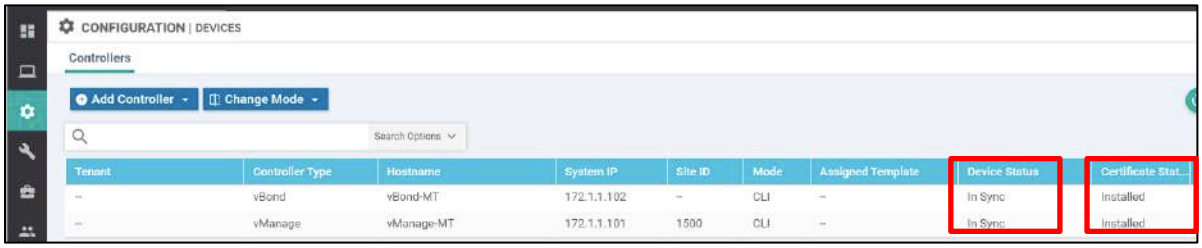
```
vBond-MT# show orchestrator connections

PEER                      PEER
         PEER     PEER     PEER             SITE        DOMAIN      PEER
PRIVATE  PEER             PUBLIC                                   ORGANIZATION
INSTANCE TYPE     PROTOCOL SYSTEM IP        ID          ID          PRIVATE IP
PORT     PUBLIC IP        PORT    REMOTE COLOR     STATE           NAME
UPTIME
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
---------------------------
0        vManage  dtls     169.254.10.1     1500        0           172.1.100.101
12346    172.1.100.101    12346   default          up              demo.local
0:01:58:21
0        vmanage  dtls     169.254.10.1     1500        0           172.1.100.101
12446    172.1.100.101    12446   default          up              demo.local
0:01:58:21

vBond-MT# show orchestrator valid-vsmarts

SERIAL NUMBER     ORG
```

```
-----------------------------
AEEABAE9B8CF74E8  demo.local

vBond-MT#
```

Step 9 In vManage go to Configuration → Devices.

Device status should be **In Sync**.

Certificate Status should be **Installed**.



Step 10 End of Task 3.

Task 4: Add a Tenant in vManage

In this task you will provision a vSmart controller and add a Tenant to vManage.

Activity Procedure

Complete these steps:

Step 1 First, you will need to provision a **vSmart-MT controller** as part of adding a **Tenant to vManage**. Once added, tenants can themselves manage vEdge devices for their organization.

The Provider needs to add the tenant's vSmart controllers.

Step 2 Using **SSH** log in to your **Pod's vSmart-MT controller**.

Step 3 Verify the configuration by issuing the command **"show run"**.

```
vsmart# show run
system
 host-name             vsmart
 admin-tech-on-failure
 aaa
!
  <output omitted>

omp
 no shutdown
 graceful-restart
!
vpn 0
!
vpn 512
 interface eth0
  ip address 10.82.83.103/20
```

 © 2020 NterOne Corporation

```
  no shutdown
 !
!
vsmart#
```

Step 4 Enter the following commands along with their appropriate values.

```
vsmart# config
Entering configuration mode terminal
vsmart(config)# system
vsmart(config-system)# host-name vSmart-MT
vsmart(config-system)# system-ip 172.1.1.103
vsmart(config-system)# site-id 2000
vsmart(config-system)# controller-group-id 10
vsmart(config-system)# sp-organization-name demo.local
vsmart(config-system)# organization-name NterOne1
vsmart(config-system)# vbond 172.1.100.102
vsmart(config-system)# exit
vsmart(config)# vpn 0
vsmart(config-vpn-0)# ip route 0.0.0.0/0 172.1.100.254
vsmart(config-vpn-0)# interface eth1
vsmart(config-interface-eth1)# ip address 172.1.100.103/24
vsmart(config-interface-eth1)# no shut
vsmart(config-interface-eth1)# tunnel-interface
vsmart(config-tunnel-interface)# allow-service netconf
vsmart(config-tunnel-interface)# allow-service sshd
vsmart(config-tunnel-interface)# exit
vsmart(config-interface-eth1)# exit
vsmart(config-vpn-0)# exit
vsmart(config)#
```

Step 5 Verify that your Pod's vSmart configuration is correct. Modify any values as necessary and then **commit**.

```
vsmart(config)# show configuration
system
 host-name             vSmart-MT
 system-ip             172.1.1.103
```

```
 site-id              2000
 controller-group-id  10
 sp-organization-name demo.local
 organization-name    NterOne1
 vbond 172.1.100.102
!
vpn 0
 interface eth1
  ip address 172.1.100.103/24
  tunnel-interface
   allow-service dhcp
   allow-service dns
   allow-service icmp
   no allow-service sshd
   allow-service netconf
   no allow-service ntp
   no allow-service stun
  !
  no shutdown
 !
 ip route 0.0.0.0/0 172.1.100.254
!
vsmart(config)# commit and-quit
Commit complete.
vSmart-MT#
```

Step 6 Issue the command **"show control connections".** There are no control connections as of yet. The vSmart-MT controller has to be added to vManage as part of **Tenant onboarding**.

```
vSmart-MT# show control connections
```

Step 7 Verify certificate status with the command **"show control local-properties".**

```
vSmart-MT# show control local-properties
personality                     vsmart
sp-organization-name            demo.local
organization-name               NterOne1
root-ca-chain-status            Installed

<output omitted>
```

 © 2020 NterOne Corporation

```
site-id                          2000
domain-id                        1
protocol                         dtls
tls-port                         23456
system-ip                        172.1.1.103
chassis-num/unique-id            5a9d76dc-d9f0-4a24-a8c0-df403fa26b9f
serial-num                       No certificate installed
token                            -NA-
retry-interval                   0:00:00:19
no-activity-exp-interval         0:00:00:20
dns-cache-ttl                    0:00:02:00
port-hopped                      FALSE
time-since-last-port-hop         0:00:00:00
number-vbond-peers               0
number-active-wan-interfaces     2


                                PUBLIC          PUBLIC PRIVATE
PRIVATE                                 PRIVATE
LAST
INSTANCE             INTERFACE  IPv4            PORT   IPv4
IPv6                                    PORT    VS/VM  COLOR
STATE CONNECTION
----------------------------------------------------------------------
----------------------------------------------------------------------
-----------------
0        eth1       172.1.100.103   12346  172.1.100.103   ::
12346     0/0   default           down    0:22:56:18
1        eth1       172.1.100.103   12346  172.1.100.103   ::
12346     0/0   default           down    0:22:56:18


vSmart-MT#
```

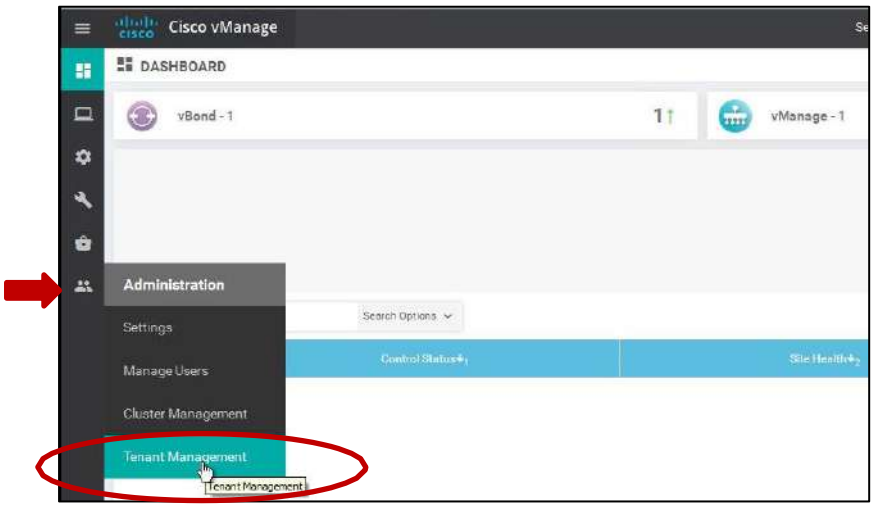
Step 8 If not already logged in, use **SSH** to log in to your **Pod's vSmart-MT Controller.**

Step 9 Now we need to move the ROOTCA.pem and ROOTCA.key files from the vManage to our newly configured vSmart. So using Putty access the vManage and scp those files over:

```
vManage-MT# vshell

vManage-MT:~$ scp ROOTCA.pem admin@172.1.100.103:

The authenticity of host '172.1.100.103 (172.1.100.103)' can't be
established.

ECDSA key fingerprint is
SHA256:2sjnjoZPpUI8g5h2LvrmN9Y67NybVD7/EXEEO0q2ddw.

Are you sure you want to continue connecting (yes/no)? yes

Warning: Permanently added '172.1.100.103' (ECDSA) to the list of known
hosts.

viptela 19.2.099


admin@172.1.100.103's password:admin

ROOTCA.pem                                    100% 1428     2.3MB/s
00:00

vManage-MT:~$ scp ROOTCA.key admin@172.1.100.103:

viptela 19.2.099


admin@172.1.100.103's password:admin

ROOTCA.key                                    100% 1675     2.2MB/s
00:00

vManage-MT:~$
```

Step 10 **Providers** will **be responsible for adding new Tenants to vManage.**

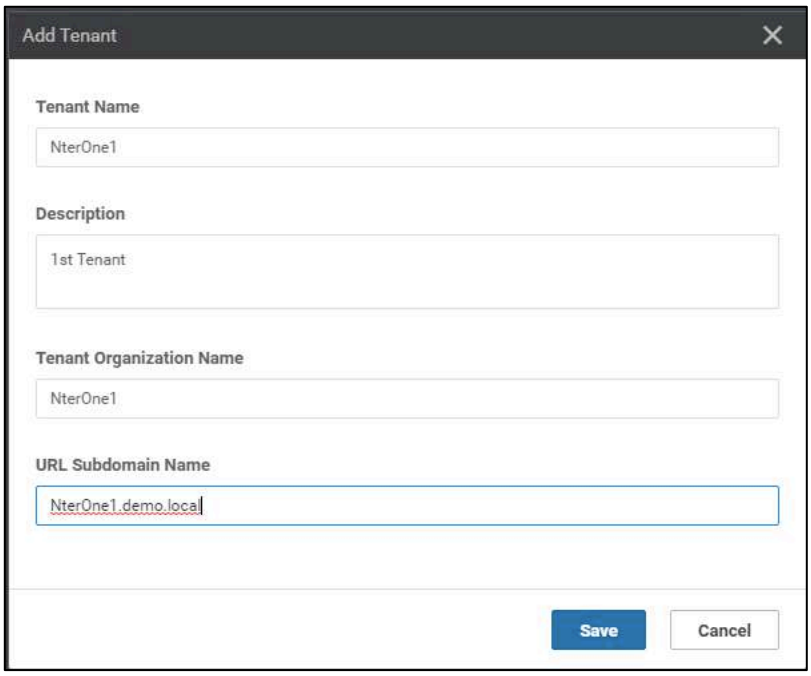
Step 11 To add tenants to vManage go to **Administration → Tenant Management**.



 © 2020 NterOne Corporation

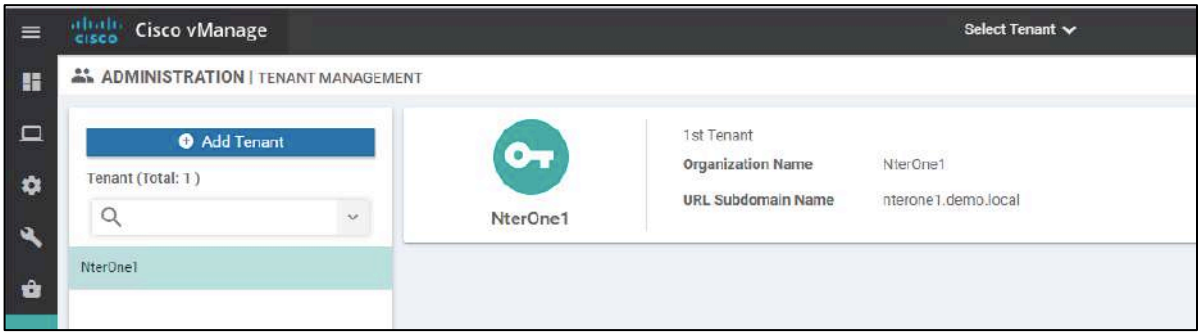
Step 12 Click **+ Add Tenant** and fill in the following values, and then click **Save**.

Tenant Name:	**NterOne1**	
Description:	**1st Tenant**	*(Required)*
Tenant Organization Name:	**NterOne1**	*(Case sensitive)*
URL Subdomain Name:	**NterOne1.demo.local**	



Step 13 You'll be redirected to **Task View**. Click the **drop-down** to view details. Verify that the **procedure is successful** or contact your Instructor if it's not.

Step 14 Go to **Administration → Tenant Management** to verify your provisioning.



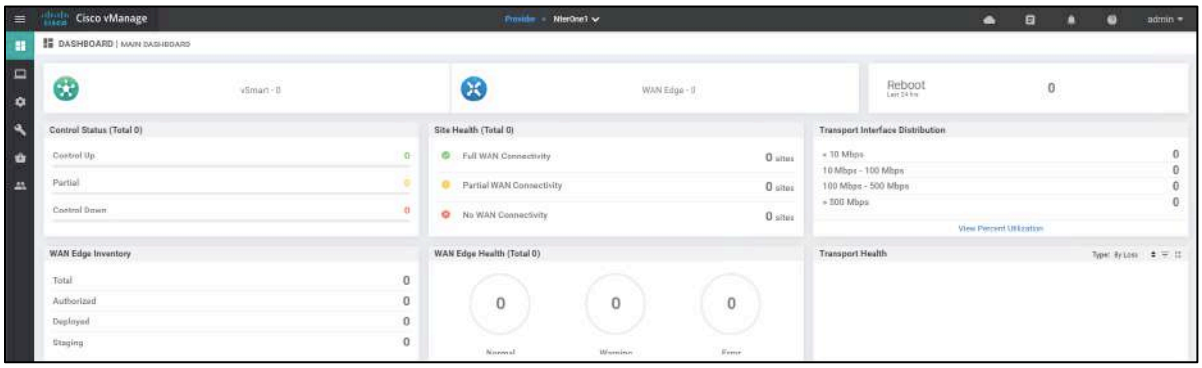
Step 15 If the Tenant data is incorrect, you can edit the tenant by clicking on the **Pencil Icon** located to the far right of the tenant entry (arrow).

Step 16 Go to vManage **Dashboard**. The dashboard window has been updated to include all tenants added to vManage.

Step 17 Click on **Select Tenant** located at top-center of the vManage window. From the drop-down list **select your new tenant**.

Step 18 You will be redirected to the **Tenant Dashboard** view. This allows the Provider to monitor and or provision elements for a selected tenant. Notice that the dashboard indicates that this is the **Provider** view of the selected **Tenant NterOne1**.

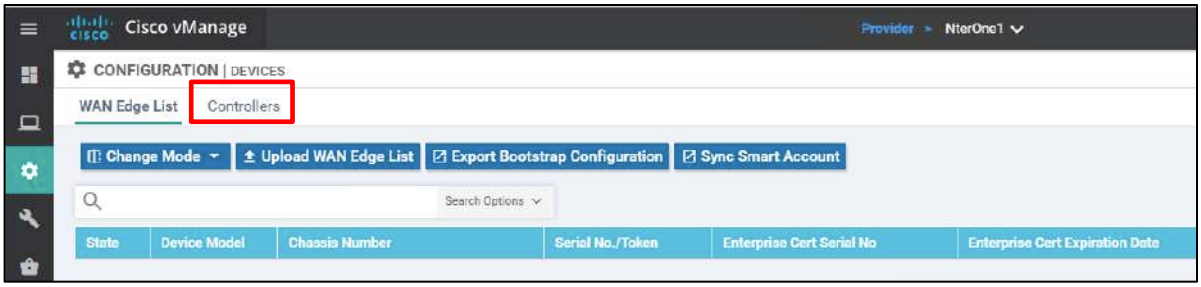
Note	**To perform the next steps DO NOT navigate away from the Provider view.**



Step 19 Go to **Configuration→ Devices**, and then click on **Controllers**.

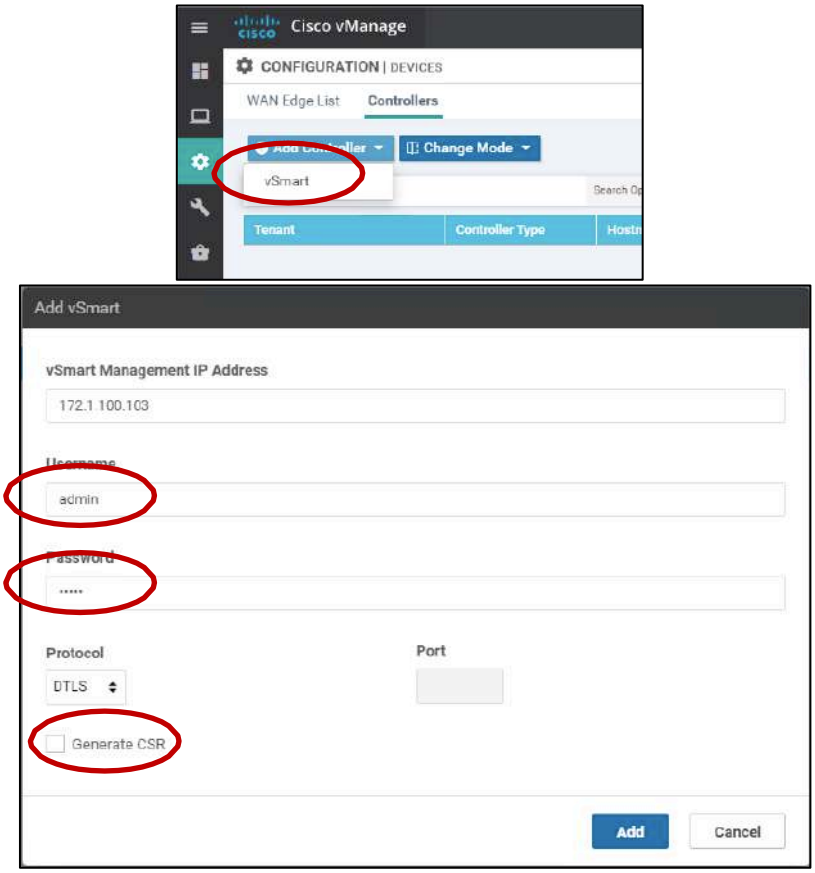
Step 20 Add **vSmart Controller** to Tenant (Provider view)

 © 2020 NterOne Corporation



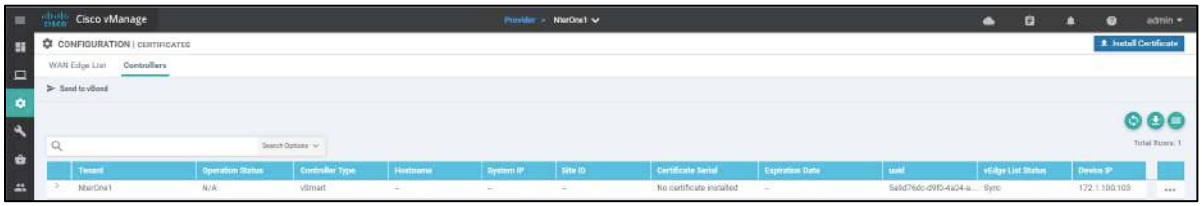
Step 21 Click **Add Controller**, and then select **vSmart**. You're going to add a vSmart controller for the selected tenant. Use the following values, and then click **Add**.

vSmart Management IP Address:	**172.1.100.103**
User/Pwd:	**admin/admin**
Generate CSR	**unchecked**

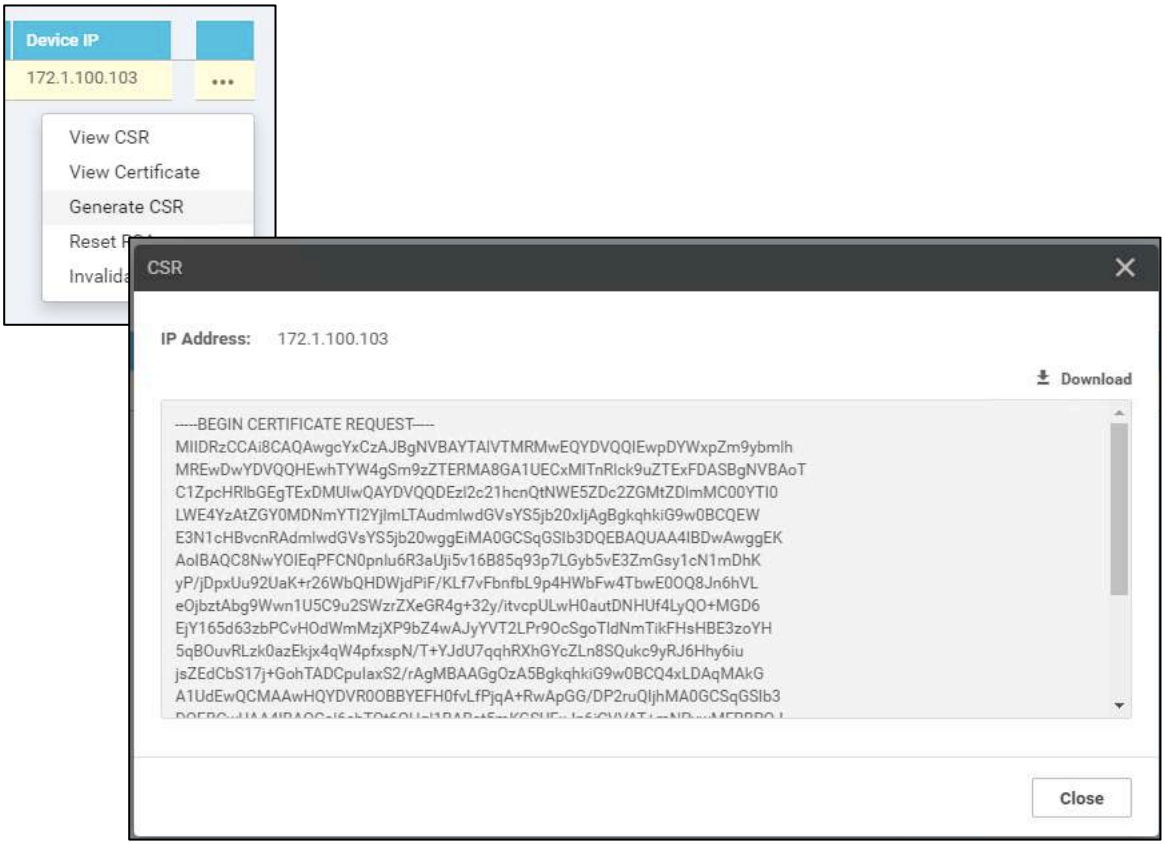


Step 22 When added successfully, the Controller will be part of the list. A certificate still needs to be installed.

Step 23 Go to **Configuration → Certificates → Controllers.**



Step 24 Scan over to the right of the window, and then click **…** and select **Generate CSR**. When the window opens **download** CSR and click **Close** when done.



Step 25 Using Putty connect to the vshell of the vSmart and verify that the vsmart_csr file is present with the other files we transferred.

```
vSmart-MT# vshell

vSmart-MT:~$ ls -l

total 44
```

 © 2020 NterOne Corporation

```
-rw-r--r-- 1 admin admin  1675 Dec 10 16:34 ROOTCA.key
-rw-r--r-- 1 admin admin  1428 Dec 10 16:34 ROOTCA.pem
-rw-r--r-- 1 admin admin   393 Dec  9 17:28 archive_id_rsa.pub
-rwxr-xr-x 1 admin admin 24736 Dec 10 16:59 master_root.crt
-rw-r--r-- 1 root  root   1212 Dec 10 17:02 vsmart_csr
vSmart-MT:~$
```

As we did with the vManage and the vBond before it we will now sign the certificate using the ROOTCA files and then copy paste that information into the vManage NMS User Interface for this Tenant (NterOne1) Note: **vsmart_csr** below.

```
vSmart-MT:~$  openssl x509 -req -in vsmart_csr -CA ROOTCA.pem -CAkey
ROOTCA.key -CAcreateserial -out vsmart.crt -days 2000 -sha256
```

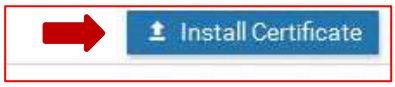
```
Signature ok
subject=/C=US/ST=California/L=San Jose/OU=NterOne1/O=Viptela
LLC/CN=vsmart-5a9d76dc-d9f0-4a24-a8c0-df403fa26b9f-
0.viptela.com/emailAddress=support@viptela.com
Getting CA Private Key
```

```
vSmart-MT:~$ cat vsmart.crt
-----BEGIN CERTIFICATE REQUEST-----
MIIDRzCCAi8CAQAwgcYxCzAJBgNVBAYTAlVTMRMwEQYDVQQIEwpDYWxpZm9ybmlh
MREwDwYDVQQHEwhTYW4gSm9zZTERMA8GA1UECxMITnRlck9uZTExFDASBgNVBAoT
C1ZpcHRlbGEgTExDMUIwQAYDVQQDEzl2c21hcnQtNWE5ZDc2ZGMtZDlmMC00YTI0
LWE4YzAtZGY0MDNmYTI2YjlmLTAudmlwdGVsYS5jb20xIjAgBgkqhkiG9w0BCQEW
E3N1cHBvcnRAdmlwdGVsYS5jb20wggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEK
AoIBAQC8NwYOIEqPFCN0pnlu6R3aUji5v16B85q93p7LGyb5vE3ZmGsy1cN1mDhK
yP/jDpxUu92UaK+r26WbQHDWjdPiF/KLf7vFbnfbL9p4HWbFw4TbwE0OQ8Jn6hVL
eOjbztAbg9Wwn1U5C9u2SWzrZXeGR4g+32y/itvcpULwH0autDNHUf4LyQO+MGD6
EjY165d63zbPCvHOdWmMzjXP9bZ4wAJyYVT2LPr9OcSgoTldNmTikFHsHBE3zoYH
5qBOuvRLzk0azEkjx4qW4pfxspN/T+YJdU7qqhRXhGYcZLn8SQukc9yRJ6Hhy6iu
jsZEdCbS17j+GohTADCpuIaxS2/rAgMBAAGgOzA5BgkqhkiG9w0BCQ4xLDAqMAkG
A1UdEwQCMAAwHQYDVR0OBBYEFH0fvLfPjqA+RwApGG/DP2ruQIjhMA0GCSqGSIb3
DQEBCwUAA4IBAQCal6chTOt6QHgl1BABst5mKGSUExJn6iCVVAT+mNPvwMFRBRQJ
67O0mJP5+taX27rS+pL844RjDT2HmOEfGZLizaQi3VAC26Cj3oaILxERQcucDMU9
+5TiYYtN1lSzS90NNHEIwZBXo3Eg1wO+KlZQnjckyUWEgJBAuJs61ohlomOSKpZi
x1osLPVY25emOEkBvuDWiqptw1rdR40ZWYHaTMBTkMBilUPWSUBnKZviMiFxNAh5
Mvbe+L9e6Vu8B9PdkMQZjOYGS8UOkGYqjeK2klKmTFJeHrIjxRLDGALBKhS49dyi
```

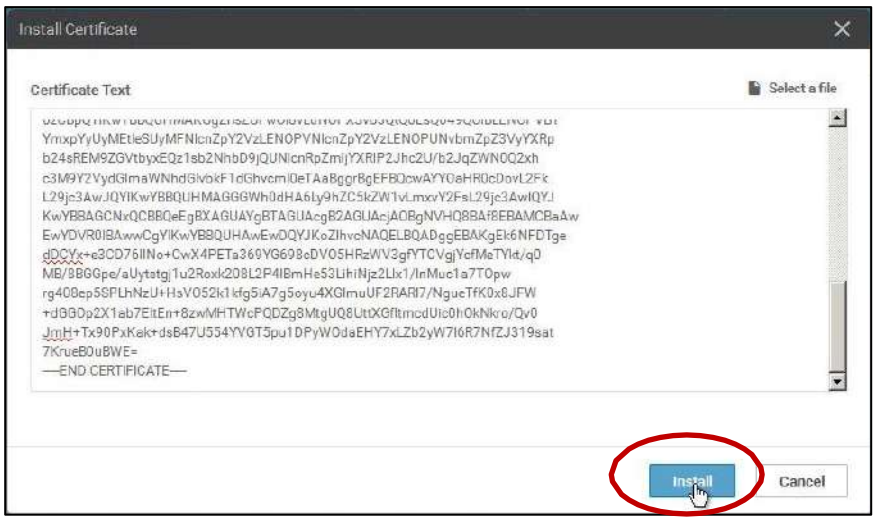
```
YKRmBTEXXFKdrnokpauhepF3Rbzb5p+YByeI
-----END CERTIFICATE REQUEST-----
vSmart-MT:~$
```

Step 26 In vManage go back to **Configuration → Certificates**.

Step 27 At the **top-right** of the window select **Install Certificate**.



Step 28 Paste the certificate you copied and click Install.



Step 29 You'll be redirected to **Task View** where you can verify the certificate installation process. Click the **drop- down** to view details. Confirm that the installation was successful or contact your Instructor if it was not.

Step 30 Go to **Configuration → Certificates → Controllers** to confirm the status of the controller. Notice the **Device IP assigned by vManage.**

Tenant	Operation Status	Controller Type	Hostname	System IP	Site ID
NterOne1	vBond Updated	vSmart	vSmart-MT	172.1.1.103	2000

Certificate Serial	Expiration Date	uuid	vEdge List Status	Device IP	
FABEF8731A977F49	01 Jun 2025 10:12:11 AM PDT	5a9d76dc-d9f0-4a24-a...	Sync	169.254.10.4	•••

Step 31 Use **SSH** to log in to your **Pod's vBond-MT.**

 © 2020 NterOne Corporation

Step 32 Issue the following commands to verify Control-Plane connectivity.

The **System IP** is assigned by vManage.

The **Organization** should match the previously provisioned value **NterOne Lab.**

The **"show orchestrator valid-vsmarts"** command should reference the vSmart certificate serial no.

```
vBond-MT# show orchestrator valid-vsmarts


SERIAL NUMBER     ORG

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

AEEABAE9B8CF74E8  demo.local

FABEF8731A977F49  NterOne1  (The vSmart Controller is added to vBond)


Bond-MT# show orchestrator connections


PEER                      PEER

         PEER     PEER     PEER             SITE        DOMAIN      PEER
PRIVATE  PEER             PUBLIC                                   ORGANIZATION

INSTANCE TYPE     PROTOCOL SYSTEM IP        ID          ID          PRIVATE IP
PORT     PUBLIC IP        PORT    REMOTE COLOR     STATE           NAME
UPTIME

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

0        vsmart   dtls     169.254.10.4     2000        1           172.1.100.103
12346    172.1.100.103    12346   default          up              NterOne1
0:00:30:09

0        vsmart   dtls     169.254.10.4     2000        1           172.1.100.103
12446    172.1.100.103    12446   default          up              NterOne1
0:00:30:09

0        vmanage  dtls     169.254.10.1     1500        0           172.1.100.101
12346    172.1.100.101    12346   default          up              demo.local
0:03:33:23

0        vmanage  dtls     169.254.10.1     1500        0           172.1.100.101
12446    172.1.100.101    12446   default          up              demo.local
0:03:33:24
```

Step 33 Go to **Administration → Manage Users**. A new user can be added so that vManage can be managed from the tenant's point of view. Click **Add User** (This is OPTIONAL).

For the remainder of this lab, the default tenant user **tenantadmin/tenantadmin** will be used.

Step 34 Log out of vManage as the Provider.

Step 35 End of Task 4.

 © 2020 NterOne Corporation

Task 5: Access vManage as a Tenant

In this Task you will access vManage as a Tenant and upload the tenant's vEdge Device list.

Activity Procedure

Complete these steps:

Step 1 **If not already done, Log out of vManage-MT as the Provider and log back in as a Tenant**. This is where the tenant URL will be utilized to determine the specific tenant logging in. **Make sure to login to the correct tenant**.

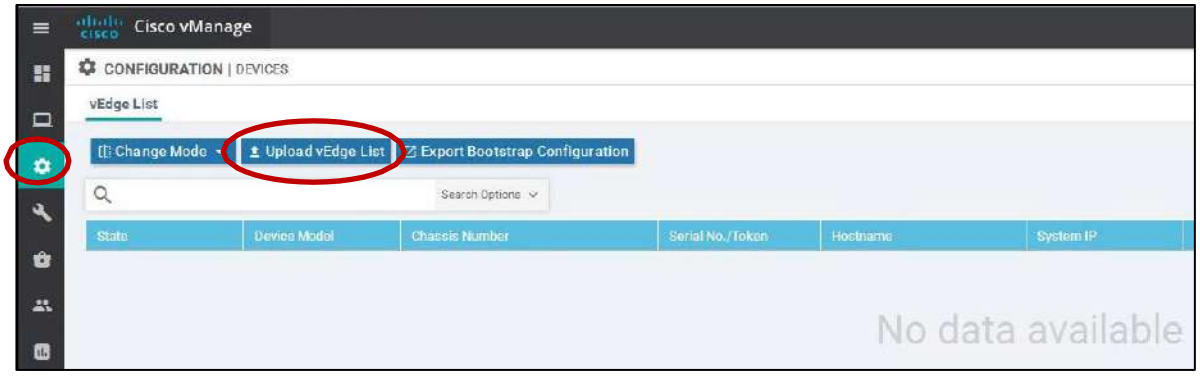
Use the following values.

- URL: https://nterone1.demo.local
- User/Pwd: tenantadmin/tenantadmin

Step 2 Once logged in, you'll be redirected to the Dashboard window. From here the tenant can monitor/manage their organizations vSmart controllers and vEdge devices. Notice that the dashboard contains only vSmart and vEdge devices. **vBond devices in Multitenant mode are managed by the Provider**.

Notice that you see the vSmart controller added from the previous task. Currently, there are no vEdge devices associated to the tenant. You're going to **upload a vEdge Device list** so that vEdge devices can be added.

Step 3 Go to **Configuration → Devices**, and then select **Upload vEdge List**.



Step 4 Browse to the folder **C:\SD-WAN** and select the file **SerialFile-19-2-MT.viptela**. Select the checkbox next to **validate the uploaded vEdge List and send to controllers**, and then click **Upload**.

Step 5 Confirm the download by clicking **OK**.

Step 6 You should receive a message indicating that the **vEdge list uploaded successfully**.

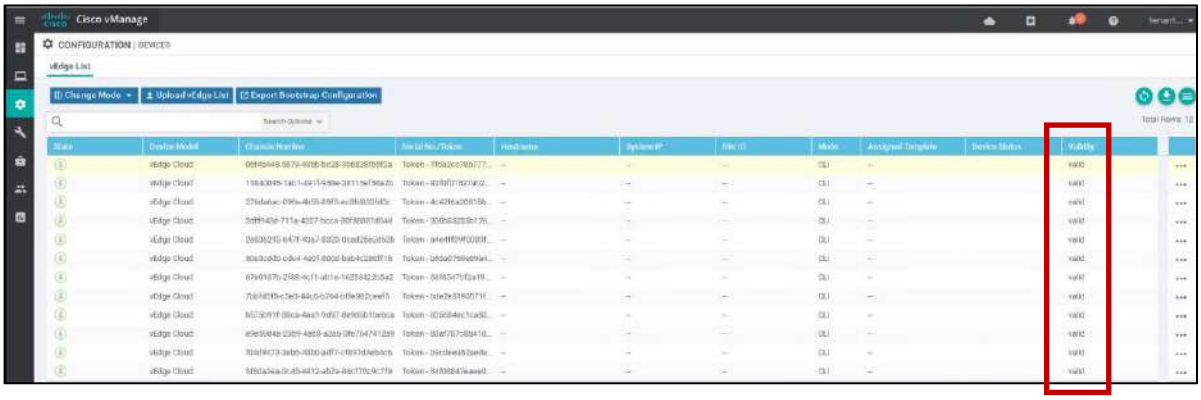
Click **OK** to close.

Contact your Instructor if you've received an error.

 © 2020 NterOne Corporation

Step 7 You'll be redirected to Task View to confirm **synchronization with the controllers**.

Step 8 Go to **Configuration → Devices**. The vEdge devices that were uploaded will be displayed here.



Step 9 Using **SSH** login to your **Pod's vBond** and determine if the vEdge devices have been added as **valid devices**.

```
vBond-MT# show orchestrator valid-vedges


HARDWARE

INSTALLED

SERIAL
CHASSIS NUMBER                          SERIAL NUMBER                      VALIDITY
ORG        NUMBER
-----------------------------------------------------------------------------------
---------------------
06F4B448-5879-43BB-BC28-3B6828FB5F2A  6adaa381a32835f186e526e1b4ba99df  valid
NterOne1   N/A
1184D095-1AB1-491F-958E-38115EF58A2B  323293c6459c9316994d71bc897875bd  valid
NterOne1   N/A
276DA6AC-09FE-4B55-89F3-EC8FD833FD0C  c6813cd999f24baf05aa5c6ea0b38059  valid
NterOne1   N/A
2DFF943D-711A-4327-BCCA-80F88881D04D  1b88e31a675801467ea15b0810ea98e9  valid
NterOne1   N/A
2E836293-647F-43A7-8323-0CAD25E2D62B  ff0afcbfc60b231ee5fa4172a9bbebd5  valid
NterOne1   N/A
```

```
50E3CDD0-CDE4-4E0F-800D-BAB4C288FF15  ca19945580ecaa80e0447e16a91aef2c  valid
NterOne1  N/A
67E0137B-2F88-4CF1-AB1E-16253422B5A2  d195a89de37a87bb495b16346853e2c9  valid
NterOne1  N/A
7BBFD5F5-C5E3-44C6-B764-BFFE982CEEF5  ed265b274b7e4dbcfbcea4792c07445d  valid
NterOne1  N/A
B575B91F-88CA-4AA1-9D97-8E905B1BEBCA  cf1c56afcaff7f2952f3cff808eb5a33  valid
NterOne1  N/A
E9E59D4A-23B9-4EB3-A2A6-0FE7647412A9  766a1657ed87de44e79aaf9a6332973c  valid
NterOne1  N/A
F04D9473-3EB6-48B0-ADF7-CF891D4EB4C6  966d429f3f10a2e28ef13f461be96e06  valid
NterOne1  N/A
FD8DA2EA-0C45-4412-AB7A-84CF70C9C7F9  9b74246f47c2fff4b3190fa8b2c259eb  valid
NterOne1  N/A

vBond-MT#
```

End of Task 5.

End of Lab 15.

All Labs completed. Congratulations!!!!!!!!

 © 2020 NterOne Corporation

Appendix A

Lab A1: Decommission a cEdge from vManage Inventory

In this Lab, you will decommission a cEdge device from the vManage inventory.

Activity Procedure

Complete these steps:

Step 1 The following vManage device will be used for this Lab.

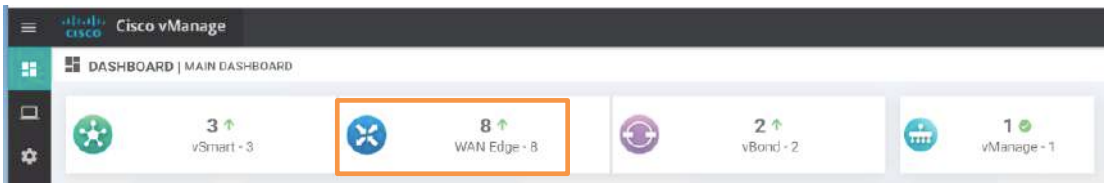
Device Name	IP Address or URL	Username	Password	LAB
vManage	https://1.4.28.28 – SSH 10.82.83.131	admin	admin	NterOne Lab

Step 2 Open a browser window and log in to vManage.

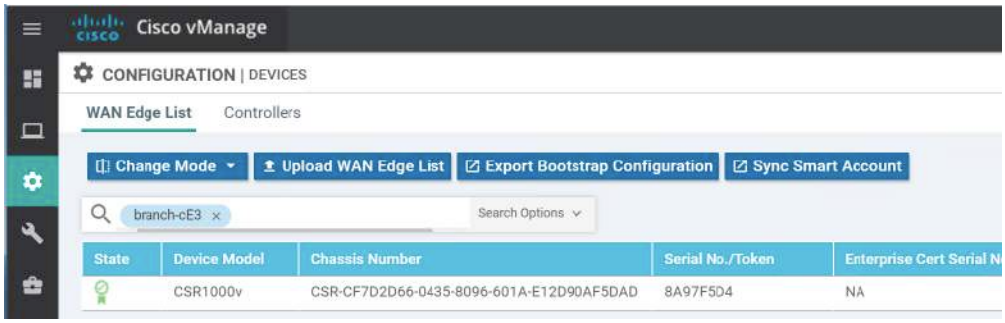
Step 3 Once logged in you will be redirected to the Dashboard.

Please take the time to observe that you have all 8 WAN Edge devices registered and operational in the current vManage Configuration.

If you do not, please notify your Instructor Immediately.



Step 4 Navigate to **Configuration > Devices.** Once you see the WAN Edge List Panel please type **branch-cE3** in the search window and press **enter** to narrow the results:



Please make you write down/or capture the Chassis Number of this specific CSR1000v because we will need this information when it comes time to add this device back to the system inventory.

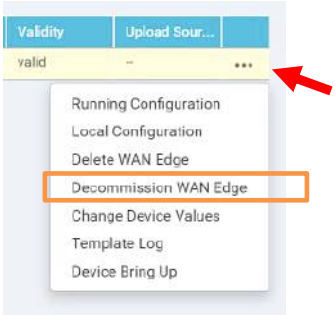
Step 5 In this lab we will employ the decommission feature of the vManage. Decommissioning can be done for both cEdge and vEdge devices and presents a fare more streamlined approach to managing available devices in the vManage NMS. We will use the slider bar to navigate completely to the right of this table so that we can see the following information:

Hostname	System IP	Site ID	Mode	Assigned Template	Device Status	Validity	Upload Sour...	
branch-cE3	172.1.30.3	30	vManage	Branch-cE3-Template	In Sync	valid	--	...

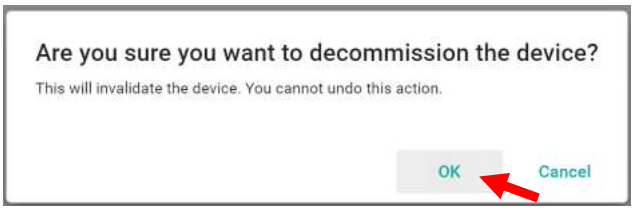
Make note that the device named **branch-cE3** is in vManage Mode, and it is attached to a template named Branch-cE3-Template and it is currently valid.

 © 2020 NterOne Corporation

Step 6 We are going to select the **…** and from the dropdown menu we will select Decommission WAN Edge.



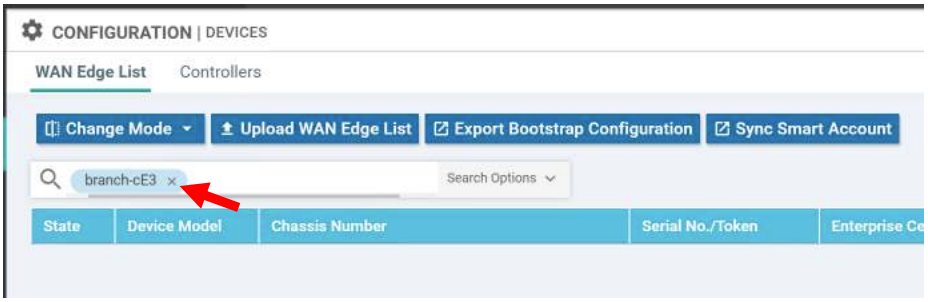
Step 7 On the popup window select OK invalidate the device:



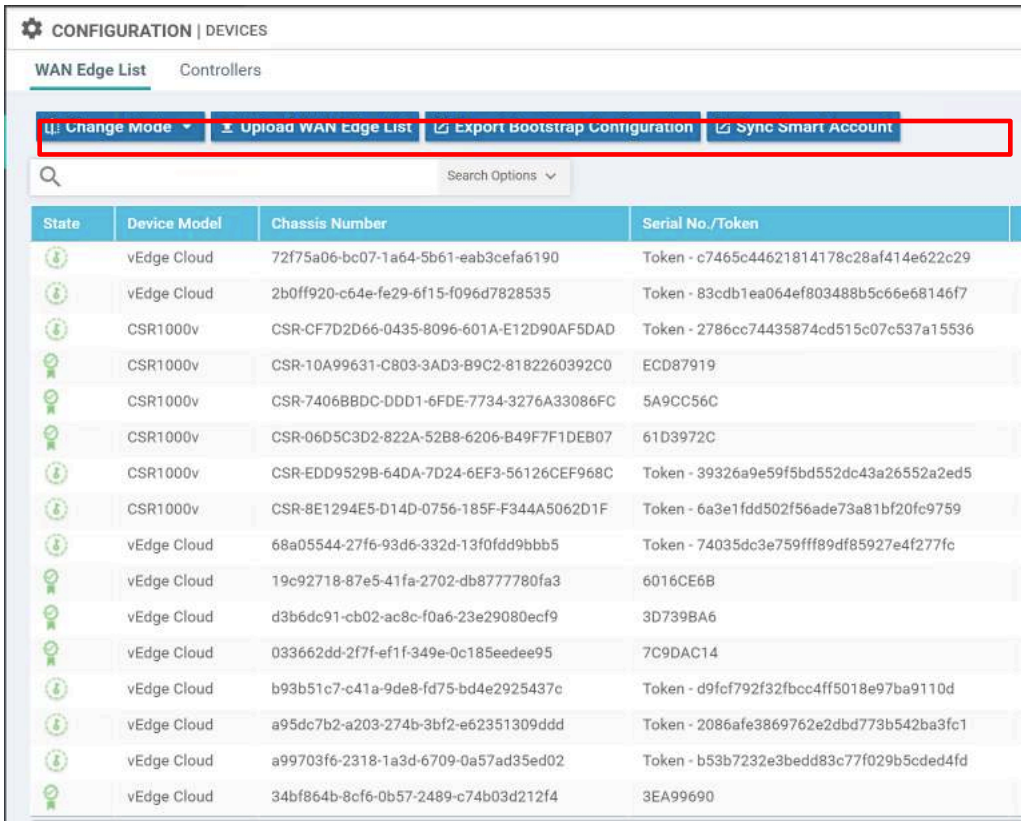
Observe carefully and you will see that the cEdge router is still listed, but only by its chassis number and now all other information is gone. If you recall in the previous labs, we invalidated, deleted a device and then reuploaded the viptela serial file. Now, we do not have to do those things using the decommission option.

We will explore that now.

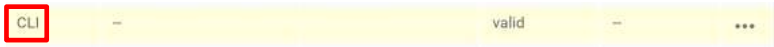
Step 8 Click the small <x> in the blue file next to branch-cE3 in the search pane:



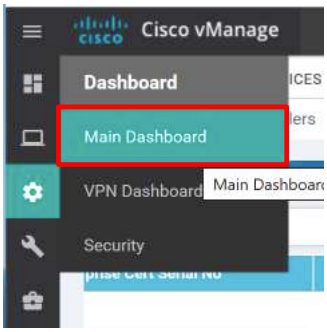
This will result in removing the search filter and we will see all the device that are defined inside the vManage NMS. You will note that the Chassis Number we made note is indeed still here, the Serial Number we had before has been removed and replace with a new OTP/Token value (your value may differ from what is illustrated here):



Step 9 Drag the slider bar completely to the right and we will see that all the previous information about this device has indeed been removed. The device is listed in CLI mode.

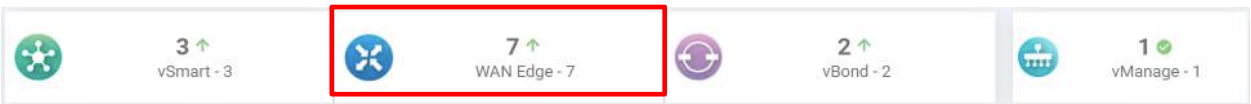


Step 10 Now we will look at the main dashboard by selecting **Dashboard > Main Dashboard**:



Step 11 Observe that we are now missing a device illustrating that decommissioning devices represents a significant efficiency over the previous method we employed.

 © 2020 NterOne Corporation



Step 12 In the next lab we will add this cEdge Device back as a serviceable device.

Activity Verification

You have completed this task when you attain these results:

- You have successfully decommissioned a cEdge device from vManage inventory.

Lab A2: Adding a cEdge to vManage Inventory

In this Lab, you will on-board the cEdge device to the vManage inventory.

Activity Procedure

Complete these steps:

Step 13 The following vManage device will be used for this Lab.

Device Name	IP Address or URL	Username	Password	LAB
vManage	https://1.4.28.28 – SSH 10.82.83.131	admin	admin	NterOne Lab

Step 14 Open a browser window and log in to vManage.

Step 15 Once logged in you will be redirected to the Dashboard.

Please take the time of observe that you have all 7 WAN Edge devices registered and operational in the current vManage Configuration.

If you do not, please notify your Instructor Immediately.

Step 16 We will now add the device we removed in the previous lab back to the vManage as a registered resource. To accomplish this, we will need to add information to the list we started building in the previous lab.

 © 2020 NterOne Corporation

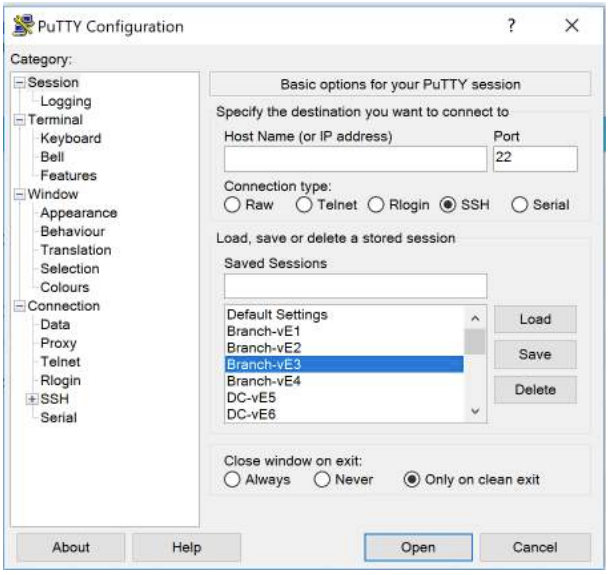
Step 17 Remember we recorded the Chassis Number of the device we are working with:

CSR-CF7D2D66-0435-8096-601A-E12D90AF5DAD

Now we will need to annotate the new OTP Token that was generated when we decommissioned the device (again your **otp** value will be different to that list)

2786cc74435874cd515c07c537a15536

Step 18 Now we will open a Putty Session to the VPN512 address of branch-cE3. This can be accomplished by using the provided Putty shortcut or



10.82.83.113:

Step 19 Once we have a session to the device, we will need to do the following:

Initially all I want to do is verify that the device is not registered or managed by the vManage NMS. This can be done by executing the following commands:

```
branch-cE3#show sdwan control connections
```

```
branch-cE3#
```

This indicates that there are no active connections between this device and the vManage, the vBond or the vSmarts. Currently it is an independent device that is officially no part of the Secure Extensible Network we have

created. We want to change that, but I want to take the opportunity to drive this point home further.

We know that for a device to be part of the our SDWAN fabric we need a valid certificate to be installed. I would like to illustrate that this device still has its certificate installed. I will do this by executing the following command on **branch-cE3**:

```
branch-cE3#show sdwan control local-properties
personality                     vedge
sp-organization-name            NterOne Lab
organization-name               NterOne Lab
root-ca-chain-status            Installed

certificate-status              Installed
certificate-validity            Valid
certificate-not-valid-before    Dec 07 14:34:11 2019 GMT
certificate-not-valid-after     Dec 04 14:34:11 2029 GMT
```

I want to illustrate that is Certificate will no long be used once we add this device back to the vManage NMS. To illustrate this, I want to look at the contents of the certificate itself. We will do this from the command line.

```
branch-cE3#show sdwan certificate installed
Installed device certificates
-----------------------------
Certificate:
    Data:
        Version: 1 (0x0)
        Serial Number: 530497183 (0x1f9ebe9f)
    Signature Algorithm: sha256WithRSAEncryption
        Issuer: C=US, CN=249dbae3-f45f-49e8-b3ef-178d17ac9a2d,
O=Viptela
        Validity
            Not Before: Dec  7 16:56:15 2019 GMT
            Not After : Dec  4 16:56:15 2029 GMT
```

 © 2020 NterOne Corporation

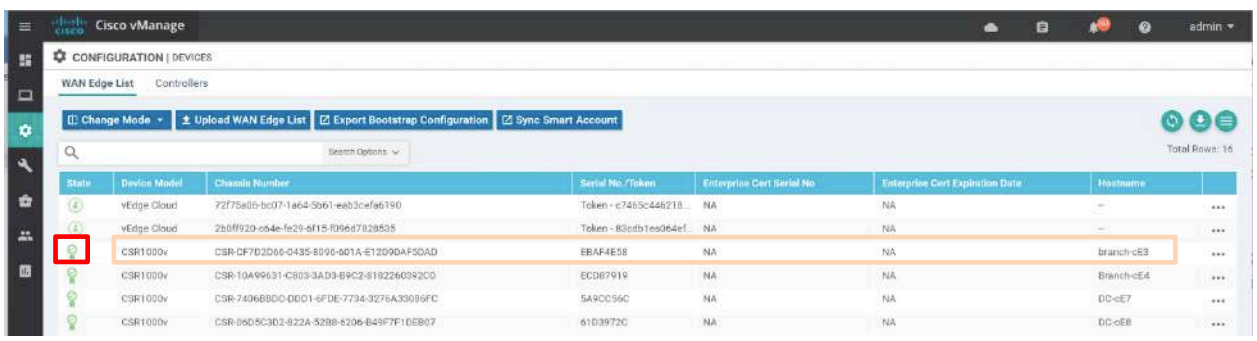
```
        Subject: C=US, ST=California, L=San Jose, OU=NterOne Lab,
O=Viptela LLC, CN=vedge-CSR-CF7D2D66-0435-8096-601A-E12D90AF5DAD-
4.viptela.com/emailAddress=support@viptela.com
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (2048 bit)
                Modulus:
                    00:a3:c4:83:fb:88:51:e9:a9:43:ed:55:6a:55:3f:
                    59:df:6f:ff:64:4b:d0:f6:6e:8c:0c:03:57:bf:50:
                    8c:ff:af:60:dd:9a:cf:36:70:b4:64:53:26:c4:d4:
                    f3:93:65:8d:d5:45:d2:03:10:27:5a:65:3c:aa:c8:
                    a5:2a:9d:08:0b:38:67:4d:24:84:6c:e0:01:e2:76:
                    32:67:6d:d1:4a:e4:77:09:34:23:ce:6a:8e:ff:40:
                    25:58:c4:67:e6:d2:6b:fe:a7:3c:f2:57:ce:44:69:
                    c6:d1:2f:d1:7a:23:97:c7:d6:c1:9b:aa:75:55:83:
                    d6:54:3f:df:01:a7:54:c8:09:88:eb:bd:e1:37:7c:
                    fb:52:8e:d9:9c:3a:46:1d:d2:76:8f:ce:7c:3d:19:
                    52:61:37:f0:e1:26:e2:2d:3f:4a:41:9d:62:c1:d8:
                    8a:75:7f:e5:a5:06:b2:8e:9d:1c:6a:89:a5:0e:7d:
                    81:86:7b:73:75:ba:2f:e7:e6:c9:1a:cb:f2:e1:cb:
                    f3:fd:00:0f:19:a5:a1:da:6a:5a:9c:3b:9c:a3:81:
                    a1:eb:ca:07:9f:bd:81:f9:60:18:ac:d8:f9:00:fe:
                    74:53:ee:d1:12:7d:23:de:56:91:27:46:24:65:2b:
                    23:74:56:e3:77:21:61:f5:fd:1b:83:32:0f:fb:e1:
                    86:99
                Exponent: 65537 (0x10001)
    Signature Algorithm: sha256WithRSAEncryption
         a7:99:54:5e:ca:29:84:db:ff:f5:d3:ea:fc:87:04:f3:d4:30:
         7d:4d:f5:36:04:f8:ab:42:0c:fd:9c:7b:be:c7:c1:f4:2e:9f:
         04:c2:6f:ff:f2:36:d0:69:2d:8f:83:ca:8b:00:f0:3c:35:b7:
         38:56:ab:42:a7:28:0d:7f:18:51:40:c2:e7:14:66:93:17:b5:
         b7:fe:76:d2:c3:14:f8:4b:f5:ae:4c:dd:92:39:b8:df:d4:80:
         ee:d7:8f:da:d8:a4:67:c1:b3:5c:79:78:ee:b4:b2:17:58:d4:
         86:3c:0d:d8:a5:26:67:b6:ca:ee:5e:41:35:ee:1d:59:2f:6b:
         e6:16:43:eb:5a:d6:b2:03:89:a3:c4:07:43:b9:c1:46:0c:a5:
```

```
0d:30:70:30:ad:95:fc:6d:89:ee:25:70:dc:22:e8:15:95:15:
00:ca:48:7c:16:87:81:4a:7e:4a:8e:1f:cd:bf:3e:d1:90:fe:
5f:15:1e:12:43:e3:30:ca:cb:92:6c:ab:ff:2f:db:f9:cb:c9:
ed:b8:8c:65:96:8c:d1:45:e3:ab:90:e5:ac:0e:43:19:b8:95:
07:c0:02:14:26:a1:17:5e:9d:5a:3b:e6:41:5a:b2:53:a7:9a:
5d:30:ad:6a:1a:03:9e:fc:51:91:df:9c:d8:ff:23:bd:7c:cd:
b2:89:e5:dc
```

Step 20 Now that we have captured the information we need to compare, we will add the cEdge back to the vManage. This will require us to use our ssh session:

```
branch-cE3#request platform software sdwan vedge activate
chassis-number CSR-CF7D2D66-0435-8096-601A-E12D90AF5DAD token
f9de4abd451a95aafdbea4c3440e08b8
branch-cE3#
```

Step 21 Return to the **Configuration > Devices** section of the vManage NMS. Observe that after a short some time, the cEdge will appear with a state of Certificate installed as indicated by the green ribbon with a check mark.



Please note that the device has a new Serial number, and that the name branch-cE3 has been detected as a part of this process.

Step 22 Now we will look at the certificate that has been installed. We will do this from the **ssh** session to **branch-cE3**.

```
branch-cE3#show sdwan certificate installed
Installed device certificates
-----------------------------
Certificate:
    Data:
        Version: 1 (0x0)
        Serial Number: 3954134616 (0xebaf4e58)
```

 © 2020 NterOne Corporation

```
    Signature Algorithm: sha256WithRSAEncryption
        Issuer: C=US, CN=249dbae3-f45f-49e8-b3ef-178d17ac9a2d,
O=Viptela
        Validity
            Not Before: Dec  7 17:03:55 2019 GMT
            Not After : Dec  4 17:03:55 2029 GMT
        Subject: C=US, ST=California, L=San Jose, OU=NterOne
Lab, O=Viptela LLC, CN=vedge-CSR-CF7D2D66-0435-8096-601A-
E12D90AF5DAD-5.viptela.com/emailAddress=support@viptela.com
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (2048 bit)
                Modulus:

00:a3:c4:83:fb:88:51:e9:a9:43:ed:55:6a:55:3f:

59:df:6f:ff:64:4b:d0:f6:6e:8c:0c:03:57:bf:50:

8c:ff:af:60:dd:9a:cf:36:70:b4:64:53:26:c4:d4:

f3:93:65:8d:d5:45:d2:03:10:27:5a:65:3c:aa:c8:

a5:2a:9d:08:0b:38:67:4d:24:84:6c:e0:01:e2:76:

32:67:6d:d1:4a:e4:77:09:34:23:ce:6a:8e:ff:40:

25:58:c4:67:e6:d2:6b:fe:a7:3c:f2:57:ce:44:69:

c6:d1:2f:d1:7a:23:97:c7:d6:c1:9b:aa:75:55:83:

d6:54:3f:df:01:a7:54:c8:09:88:eb:bd:e1:37:7c:

fb:52:8e:d9:9c:3a:46:1d:d2:76:8f:ce:7c:3d:19:

52:61:37:f0:e1:26:e2:2d:3f:4a:41:9d:62:c1:d8:

8a:75:7f:e5:a5:06:b2:8e:9d:1c:6a:89:a5:0e:7d:

81:86:7b:73:75:ba:2f:e7:e6:c9:1a:cb:f2:e1:cb:

f3:fd:00:0f:19:a5:a1:da:6a:5a:9c:3b:9c:a3:81:

a1:eb:ca:07:9f:bd:81:f9:60:18:ac:d8:f9:00:fe:

74:53:ee:d1:12:7d:23:de:56:91:27:46:24:65:2b:

23:74:56:e3:77:21:61:f5:fd:1b:83:32:0f:fb:e1:
```

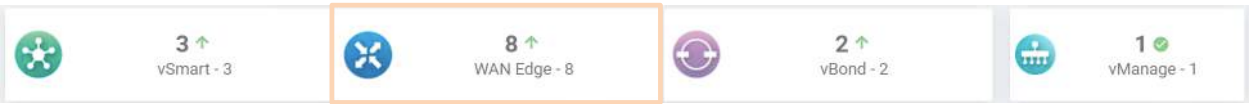
```
                    86:99
                Exponent: 65537 (0x10001)
    Signature Algorithm: sha256WithRSAEncryption

c7:c1:fb:ab:cc:03:d3:8f:4a:93:70:4e:2d:1a:b1:2b:3d:09:

c2:6c:e0:e0:10:a6:4c:d3:20:5e:3b:5e:87:46:2a:09:a6:55:

1d:8b:cc:a6:e5:1c:92:45:d3:d3:36:ab:96:6e:42:38:ba:15:

ef:46:27:ce:77:d8:95:e3:a5:d7:c1:98:24:c9:8e:fd:25:40:

9a:43:5b:4e:83:f1:20:e9:b2:1d:93:ff:e7:5c:8e:c8:e4:c3:

7d:df:ef:ca:8c:29:a1:ef:cf:d7:06:45:bd:d0:a3:21:36:23:

77:28:fb:0c:93:d5:8c:ae:d5:c2:e8:51:e7:81:13:43:2b:d6:

a8:d7:84:fb:fe:70:38:64:35:8e:5c:a7:9e:f3:79:09:a8:55:

1a:d7:c0:23:3d:7d:31:79:f4:32:56:79:c5:9e:b5:97:16:21:

43:f8:8c:6e:91:92:bb:0b:21:8f:68:02:06:ce:80:db:ac:d8:

bd:ef:0e:63:d6:6d:00:45:1e:0a:5d:ac:af:1e:e5:8a:63:cb:

ca:0f:bf:54:fb:38:df:06:ab:eb:7f:dc:ad:aa:93:59:c5:37:

25:d8:82:80:4e:37:d8:dc:ee:06:bc:d3:1a:43:e0:05:5c:d4:

7e:f6:6a:21:ee:f4:a1:a7:4c:88:a6:4d:e5:7f:5c:b9:c8:b3:
          fb:85:e0:5a
```

Observe that the current certificate is different than the previous one. Here they are side by side.

`fb:85:e0:5a` verses `b2:89:e5:dc`

This clearly illustrates that the certificate is a result of a new CSR being generated as a result of the commands we entered.

Step 23 Now we should see that we have all 8 devices installed and operational in the vManage NMS dashboard, **Dashboard > Main Dashboard**:



 © 2020 NterOne Corporation

Step 24 Make a note that this device is currently not vManaged. We will leave it this way until later in our explorations.

Activity Verification

You have completed this task when you attain these results:

- You have successfully added a cEdge device to the vManage inventory.