

CISCO

SD-WAN

Configure From Scratch

FULL LAB DOCUMENTATION

MUHAMMAD RIDHO CAHYO

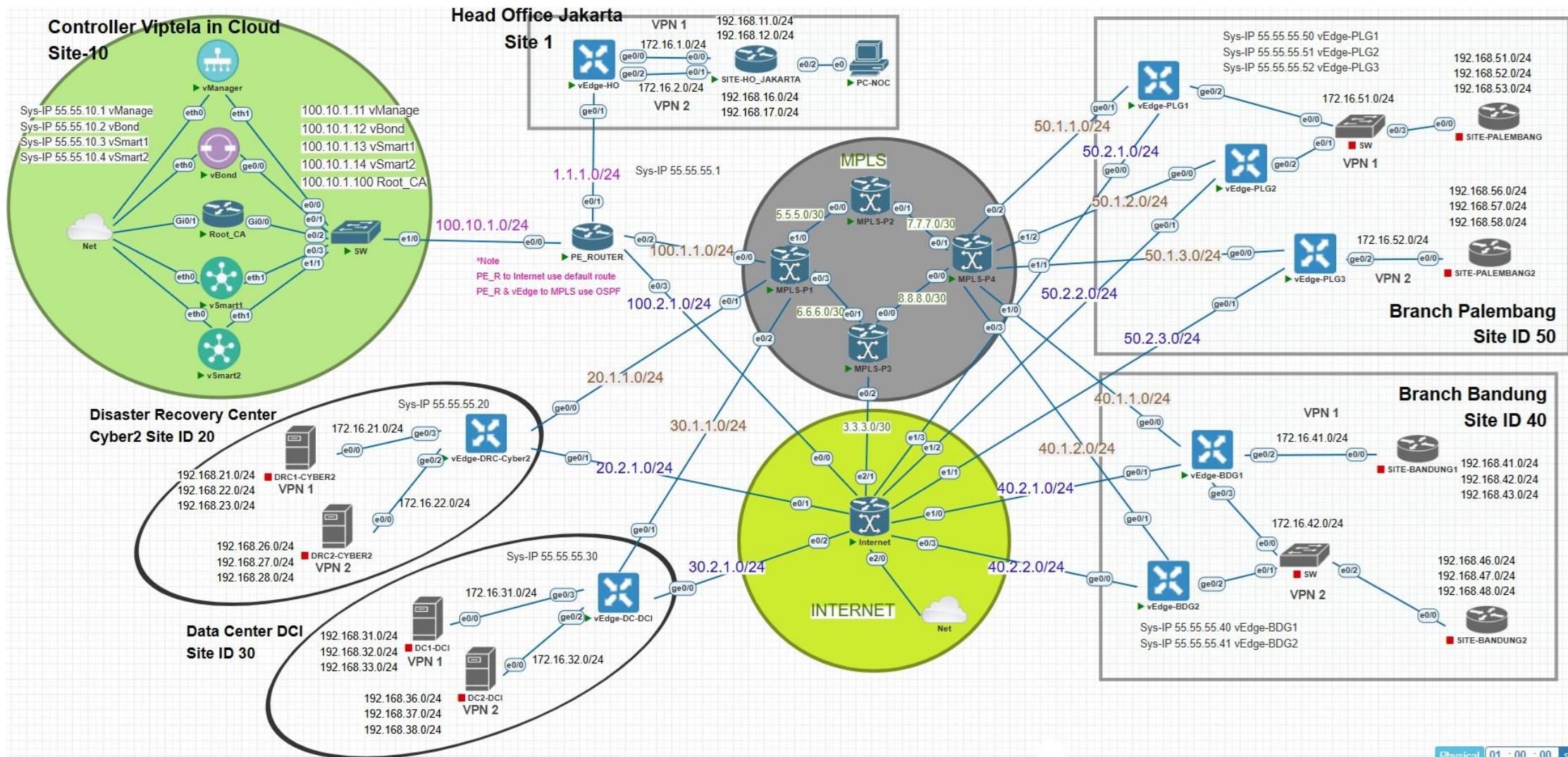
Table of Contents

TOPOLOGY	4
CONFIGURE INTERNET & MPLS ROUTER	6
PE_Router	7
MPLS Backbone	7
Internet Router.....	9
CONFIGURE SD-WAN CONTROLLER	11
vBond.....	13
vSmart	14
vManage	16
INSTALL CERTIFICATE & ONBOARDING SD-WAN CONTROLLER.....	17
Step by step:	19
1. Setup Root CA	19
2. Setup vManage Certificate: download & install root-ca cert, generate CSR	21
3. Setup vBond Certificate: download & install root-ca cert, generate CSR.....	26
4. Setup vSmart Certificate: download & install root-ca cert, generate CSR ...	30
5. Verify the onboarded controller.	34
INSTALL CERTIFICATE & ONBOARDING SD-WAN EDGE / VEDGE.....	36
Step by step:	38
1. Setup Basic Config vEdge (system, VPN0, VPN512).....	38
2. Download & Install Root Certificate from Root_CA	39
3. Generate CSR from vEdge	39
4. Request Signing to Root_CA & Paste granted certificate CSR into a new file	40
5. Add vEdge to Controller (Onboard vEdge based Chassis & Serial num)	41
6. Verifikasi Onboarded vEdge.....	42
BRING-UP MPLS TRANSPORT.....	43
Step by step:	43
1. Setup OSPF point-to-point on MPLS-P1 MPLS-P2 (link ke vEdge).....	44
2. Configure OSPF on all vEdge (for reachability to controllers through MPLS)	44
3. Verification OSPF reachability to controller.....	45

4. Connect MPLS & Internet to Bring UP Full BFD Session/Reachability	47
VMANAGE TEMPLATE OVERVIEW	49
Device template.....	49
Feature template	49
How the templates are created and deployed	51
Device Template Method.....	52
Step by step:	53
1. Create and setup Features template	53
2. Create and setup Device template	62
3. Attach Device template to Device	64
4. Verification Pushed Device Template	67
SERVICE VPN OVERVIEW: CONNECTED & STATIC ROUTES.....	68
Scenario:	70
VPN 1 : Connected Routes (CLI and Templates)	70
VPN 1 : Static Routes (CLI and Templates)	76
VPN 1 : Verification.....	80
SERVICE VPN: VRRP & DHCP	83
Scenario:	83
VPN 1 : VRRP (CLI and Templates).....	84
VPN 1 : DHCP (CLI and Templates).....	88
VPN 1 : Verification.....	91
SERVICE VPN: OSPF CONFIGURATION.....	93
Scenario:	93
Setup LAN Facing Device	93
VPN 1 : OSPF PEERING VIA CLI	94
VPN 1 : OSPF PEERING VIA TEMPLATES	95
VPN 1 : OSPF Verfication	99
SERVICE VPN: BGP CONFIGURATION	101
Scenario:	101
Setup LAN Facing Device	102
VPN 2 : iBGP PEERING & ROUTE PROPAGATION	106

VPN 2 : iBGP Verification	116
VPN 2 : eBGP PEERING & ROUTE PROPAGATION	119
VPN 2 : eBGP Verification	125
VPN 2 : BGP Localized Route Policy	130
VPN 2 : BGP Route Policy Verification	144
VSMART TEMPLATE SETUP AND OVERVIEW	146
 Centralized Policies Overview	146
 What Is Inside Centralized Policies?.....	147
 vSmart Template Setup	150
 Hub & Spoke Setup	158
REFERENCES.....	169

TOPOLOGY



In the above topology, the scenario involves an organization named 'MRC' that operates across six sites located in various on-premises and cloud environments, including:

1. Site 1: Head Office, Jakarta
2. Site 10: Controller Viptela (Cloud)
3. Site 20: Data Center Cyber2
4. Site 30: Data Center DCI
5. Site 40: Branch Office, Bandung
6. Site 50: Branch Office, Palembang

This infrastructure uses SD-WAN to manage all connections and orchestration for each site. SD-WAN separates the control-plane and data-plane functions using Cisco Viptela controllers (vManage, vBond, vSmart).

In the underlay network, each site connects to the controllers using two different transport links. One of SD-WAN's advantages is its ability to support multiple transport mediums for underlay interconnection. In this scenario, the transport links used are **Internet** and **MPLS**.

CONFIGURE INTERNET & MPLS ROUTER

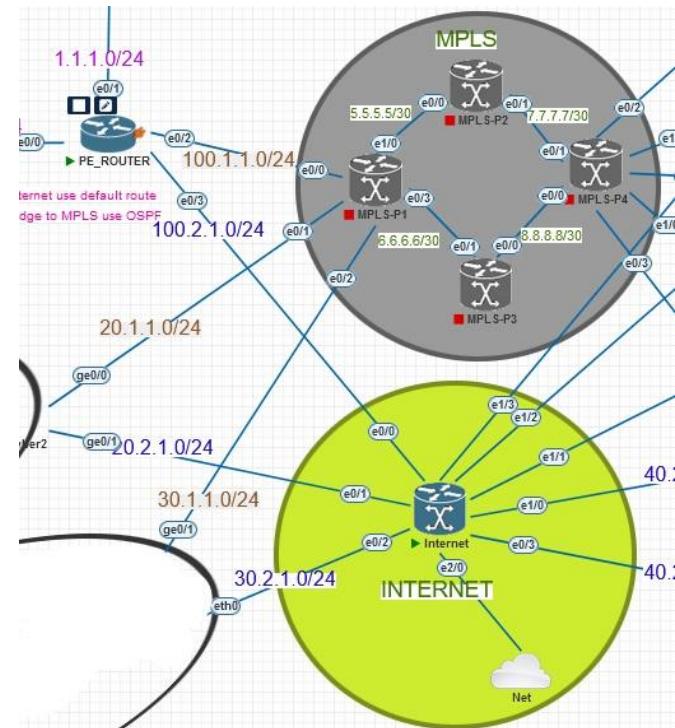
Using dual transport links in SD-WAN has several benefits, including:

1. Redundancy and High Availability
2. Load Balancing
3. Cost Efficiency
4. Traffic Segmentation

SD-WAN can flexibly utilize all its transport links. For example, in traffic segmentation, we can choose specific paths for critical applications requiring security and low latency or select other paths for non-critical/cloud access.

Configuration:

PE_Router, MPLS Backbone, Internet Router



The interconnection used for the MPLS backbone transport link is OSPF routing. For the Internet router, a default route is applied.

PE_Router

```
// Config P2P / Interface & loopback address according to the topology

// Config OSPF

router ospf 1

router-id 1.1.1.1

network 1.1.1.0 0.0.0.255 area 0

network 100.1.1.0 0.0.0.255 area 0

network 100.10.1.0 0.0.0.255 area 0

// Config MPLS in OSPF

((OSPF automatically functions as the backbone for MPLS interconnection.))

mpls ip

mpls ldp router-id lo0 force

router ospf 1

mpls ldp autoconfig

// Config Default Route to Internet router

ip route 0.0.0.0 0.0.0.0 100.2.1.2
```

MPLS Backbone

MPLS-P1

```
// Configure P2P / Interface & loopback address according to the topology

// Config OSPF (MPLS interconnection)

router ospf 1

router-id 2.2.2.1

network 0.0.0.0 255.255.255.255 area 0
```

```
// Config MPLS in OSPF  
mpls ip  
mpls ldp router-id lo0 force  
router ospf 1  
mpls ldp autoconfig
```

MPLS-P2

```
// Configure P2P / Interface & loopback address according to the topology
```

```
// Config OSPF (MPLS interconnection)  
router ospf 1  
router-id 2.2.2.2  
network 0.0.0.0 255.255.255.255 area 0
```

```
// Config MPLS in OSPF  
mpls ip  
mpls ldp router-id lo0 force  
router ospf 1  
mpls ldp autoconfig
```

MPLS-P3

```
// Configure P2P / Interface & loopback address according to the topology
```

```
// Config OSPF (MPLS interconnection)  
router ospf 1  
router-id 2.2.2.3  
network 0.0.0.0 255.255.255.255 area 0
```

```
// Config MPLS in OSPF  
mpls ip  
mpls ldp router-id lo0 force
```

```
router ospf 1
mpls ldp autoconfig
```

MPLS-P4

```
// Config P2P / Interface & loopback address according to the topology
```

```
// Config OSPF (MPLS interconnection)
```

```
router ospf 1
router-id 2.2.2.4
network 0.0.0.0 255.255.255.255 area 0

// Config MPLS in OSPF
mpls ip
mpls ldp router-id lo0 force
router ospf 1
mpls ldp autoconfig
```

Internet Router

```
// Config P2P / Interface & loopback address according to the topology
```

```
// Setup Internet
interface Ethernet2/0
description TO INTERNET PUBLIC INTERNET
ip address dhcp
end
!
```

```
# Default route to the internet gateway
ip route 0.0.0.0 0.0.0.0 192.168.18.33
!
```

```
# Config NAT
access-list 10 permit any
!
ip nat inside source list 10 interface Ethernet2/0 overload
!
interface range Ethernet0/0-3, Ethernet1/0-3
    ip nat inside
!
interface Ethernet2/0
    ip nat outside

// Static route to all Site segments
ip route 1.1.1.0 255.255.255.0 100.2.1.1
ip route 100.10.1.0 255.255.255.0 100.2.1.1
ip route 172.16.1.0 255.255.255.0 100.2.1.1
ip route 172.16.20.0 255.255.255.0 20.2.1.1
ip route 172.16.30.0 255.255.255.0 30.2.1.1
ip route 172.16.40.0 255.255.255.0 40.2.2.1
ip route 172.16.40.0 255.255.255.0 40.2.1.1
ip route 172.16.50.0 255.255.255.0 50.2.3.1
ip route 172.16.50.0 255.255.255.0 50.2.2.1
ip route 172.16.50.0 255.255.255.0 50.2.1.1
!
```

CONFIGURE SD-WAN CONTROLLER

VMANAGE, VSMART, VBOND

In this section, we will configure the controllers (vManage, vBond, vSmart). SD-WAN technology introduces the concept of VPNs to isolate traffic and provide network segmentation. These VPNs function similarly to Virtual Routing and Forwarding (VRF), meaning that each VPN has its own routing table and can be managed independently.

There are several types of VPNs in Viptela SD-WAN with different functions:

1. VPN 0 (Transport VPN)

- Used for transport connectivity between SD-WAN devices and the edge network.
- Utilizes physical interfaces/tunnels to establish connections to the WAN/Internet.
- Typically connected to WAN or transport links (MPLS, Internet, or LTE).
- All communication between SD-WAN devices through the overlay network occurs via VPN 0.
- Establishes IPSec tunnels between SD-WAN devices.
- Handles all **OMP (Overlay Management Protocol)**, **BFD (Bidirectional Forwarding Detection)**, and **DTLS/TLS** communication with vSmart, vBond, and vManage.

2. VPN 512 (Management VPN)

- Connects SD-WAN devices to the management network (SSH, SNMP, API).
- Provides out-of-band connectivity separated from data VPNs, ensuring no interference with production traffic.

3. VPN 1 – 511 (Service VPN)

- Enables traffic isolation similar to VRF.

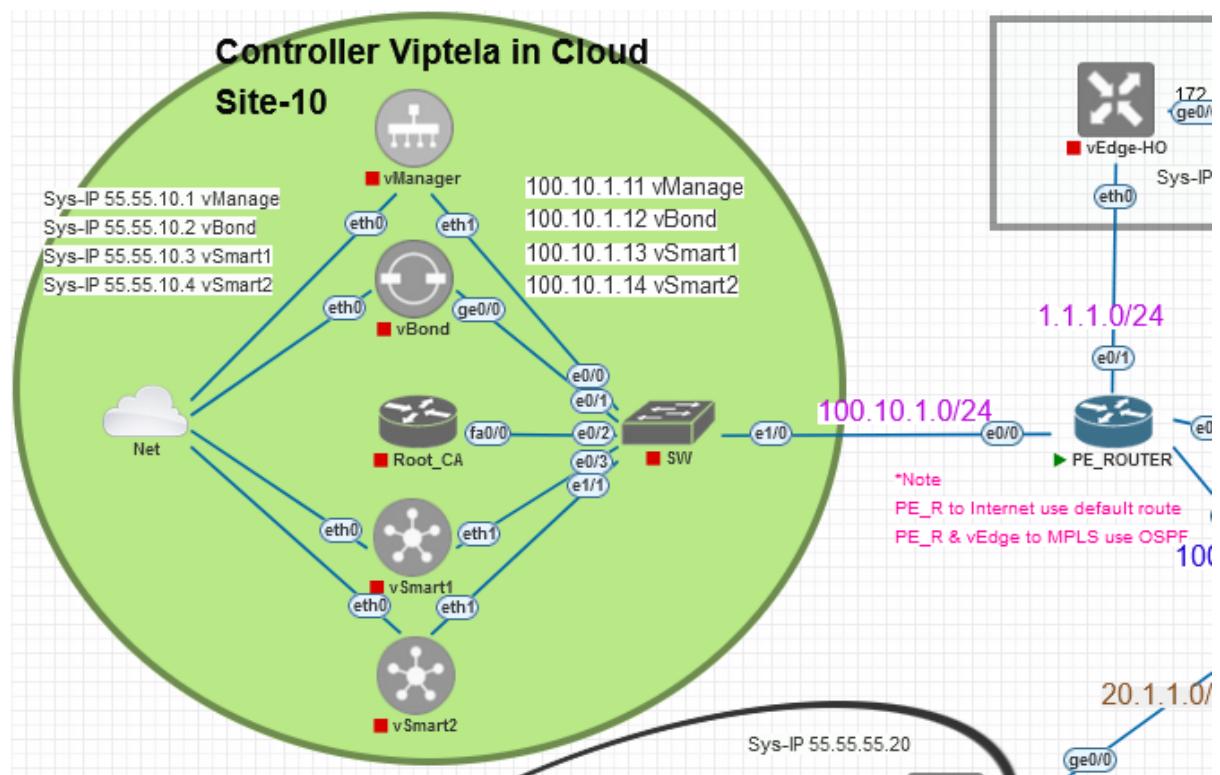
- Connects LAN networks to the SD-WAN overlay and carries customer traffic to destinations via IPSec tunnels.
- Enables independent QoS, ACL, and firewall policies for each VPN.

4. VPN 513 – 65530 (User-defined VPN)

- Functions like VPN 1–511.
- Used for additional segmentation or specific applications requiring isolation or special network connections (e.g., IoT, Hybrid Cloud, or Multi-Tenancy).

Configuration:

vBond, vSmart, & vManage



vBond

```
system
  host-name vBond
  system-ip 55.55.10.2
  site-id 10
  organization-name MRC
  clock timezone Asia/Jakarta
  vbond 100.10.1.12 local

vpn 0
  interface ge0/0
    ip address 100.10.1.12/24
    tunnel-interface
      encapsulation ipsec
      allow-service all
      allow-service sshd
      allow-service netconf
    !
    no shutdown
  !
  ip route 0.0.0.0/0 100.10.1.10
  !

vpn 512
  interface eth0
    ip dhcp-client
    ipv6 dhcp-client
    no shutdown
  !
commit
```

vSmart

```
// vSmart 1

system
  host-name vSmart1
  system-ip 55.55.10.3
  site-id 10
  organization-name MRC
  clock timezone Asia/Jakarta
  vbond 100.10.1.12

vpn 0
  interface eth1
    ip address 100.10.1.13/24
    tunnel-interface
      allow-service all
      allow-service sshd
      allow-service netconf
    !
    no shutdown
  !
  ip route 0.0.0.0/0 100.10.1.10
  !

vpn 512
  interface eth0
    ip dhcp-client
    ipv6 dhcp-client
    no shutdown
  !
  commit
```

```
// vSmart 2

system

host-name vSmart2

system-ip 55.55.10.4

site-id 10

organization-name MRC

clock timezone Asia/Jakarta

vbond 100.10.1.12


vpn 0

interface eth1

ip address 100.10.1.14/24

tunnel-interface

allow-service all

allow-service sshd

allow-service netconf

!

no shutdown

!

ip route 0.0.0.0/0 100.10.1.10

!

vpn 512

interface eth0

ip dhcp-client

ipv6 dhcp-client

no shutdown

!

commit
```

vManage

```
system
  host-name vmanage
  system-ip 55.55.10.1
  site-id 10
  organization-name mrc
  clock timezone asia/jakarta
  vbond 100.10.1.12

vpn 0
  interface eth1
    ip address 100.10.1.11/24
    tunnel-interface
      allow-service all
      allow-service sshd
      allow-service netconf
      allow-service https
    !
    no shutdown
  !
  ip route 0.0.0.0/0 100.10.1.10
  dns 8.8.8.8
  !
vpn 512
  interface eth0
    ip dhcp-client
    ipv6 dhcp-client
    no shutdown
```

ONCE THE CONFIGURATION IS COMPLETE, VMANAGE GUI CAN BE ACCESSED.

CONNECT A PC TO THE SAME SEGMENT AS VMANAGE AND ACCESS VIA A WEB BROWSER.

INSTALL CERTIFICATE & ONBOARDING SD-WAN CONTROLLER

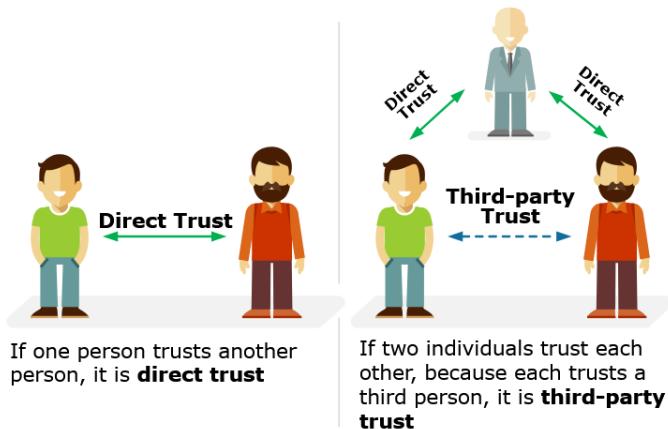
In SD-WAN, there is a process where each device must verify the digital certificates of other devices.

Why is this certificate process necessary? SD-WAN edge routers (vEdge) communicate with controllers (vManage, vBond, vSmart) using DTLS tunnels. This process is similar to TLS/SSL, requiring a handshake and involving digital certificates.

When two devices verify each other's certificates, a "trust" relationship is established (validated/authenticated). This trust can be built in two ways:

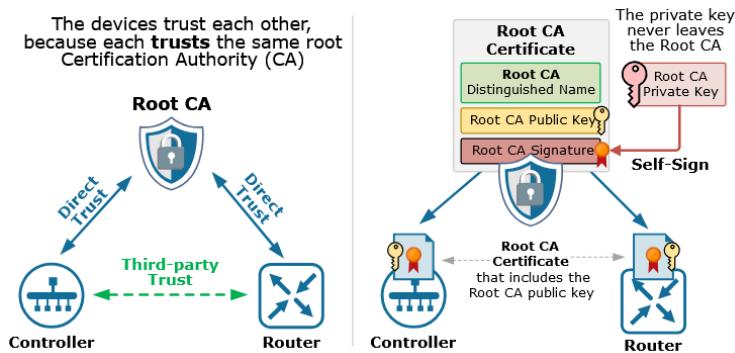
- Direct Trust: One device directly trusts another.
- Third-Party Trust: Two devices trust a third-party entity, the Certificate Authority (CA).

Due to the scalability of SD-WAN and the increasing number of WAN edges in an organization, the Direct Trust model is not feasible. Instead, SD-WAN uses the Third-Party Trust model.



Due to the vast scalability of SD-WAN and the growth of WAN Edge over time within an organization, the "Direct trust" model is not feasible. This is why SD-WAN communication uses the "Third-party trust" model. Each device will "trust" one another because they

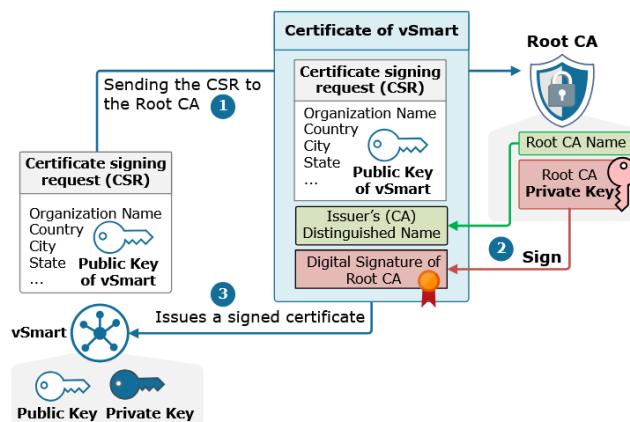
have previously trusted the same third-party entity, known as the "Certificate Authority (CA)" or "Root CA."



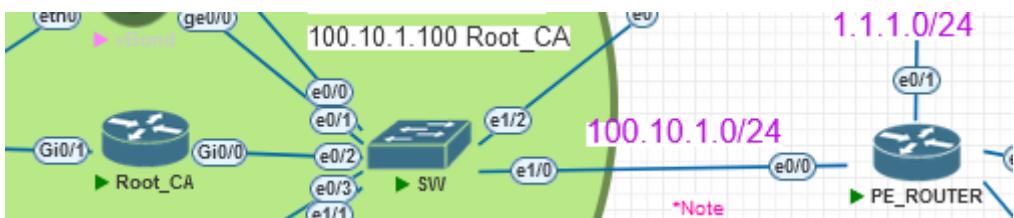
The Root_CA will issue a certificate, which we refer to as the root certificate. This certificate, owned by the Root_CA, needs to be installed on each device in order to identify the Root_CA as their third-party entity for establishing trust between devices. However, this process is one-sided (from the SD-WAN device to the Root_CA), and the Root_CA does not yet "trust" the SD-WAN device.

When a device wants to establish trust, it will initiate an SSL/TLS connection and also send a "Certificate Signing Request (CSR)" to the Root CA. The CSR contains information such as the device's name, organization, domain name, and so on, along with the device's public key.

Once the Root CA receives the CSR from the device, it verifies the information and signs the certificate with its private key. The Root CA then sends the signed certificate back to the requesting device as a "granted certificate." As a result, the certificate is now digitally signed, and any device with a certificate from the same Root_CA can verify the signature.



Step by step:



*image Root_CA: [QEMU] vios-adventerprisek9-m-15.6.2T

1. Setup Root CA

```
// Config IP dan default route
IOS-Root-CA(config)# int g0/0
IOS-Root-CA(config-if)# ip add 100.10.1.100 255.255.255.0
IOS-Root-CA(config-if)# no sh
!
IOS-Root-CA(config)# int g0/1
IOS-Root-CA(config-if)# ip add dhcp
IOS-Root-CA(config-if)# no sh
!
IOS-Root-CA(config)# ip route 0.0.0.0 0.0.0.0 100.10.1.10      << pe_router

// Create a pair of RSA public-key and private-key.
Router(config)# hostname IOS-Root-CA
IOS-Root-CA(config)# crypto key generate rsa label PKI modulus 2048
The name for the keys will be: PKI
% The key modulus size is 2048 bits
% Generating 2048 bit RSA keys, keys will be non-exportable...
[OK] (elapsed time was 6 seconds)

*Nov 6 07:49:33.179: %SSH-5-ENABLED: SSH 1.99 has been enabled

// Activate the web server for GUI access in certificate management.
IOS-Root-CA(config)# ip http server
```

```
// Create a CA server for issuing digital certificates. The CA server functions to
manage digital certificates, such as creating, signing, or revoking certificates.

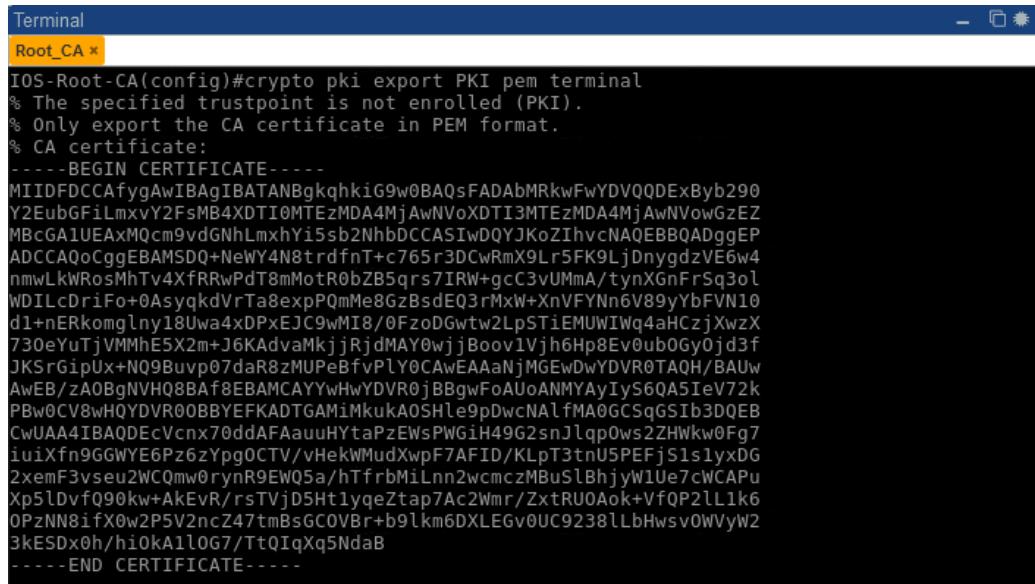
IOS-Root-CA(config)# crypto pki server PKI
IOS-Root-CA(cs-server)# database url flash:
IOS-Root-CA(cs-server)# database level complete
IOS-Root-CA(cs-server)# issuer-name cn=rootca.lab.local
IOS-Root-CA(cs-server)# hash sha256
IOS-Root-CA(cs-server)# database archive pkcs12 password cisco123
IOS-Root-CA(cs-server)# grant auto
*Mar 1 00:13:16.419: %PKI-6-CS_GRANT_AUTO: All enrollment requests will be
automatically granted.

IOS-Root-CA(cs-server)# no shut
%Some server settings cannot be changed after CA certificate generation.
% Certificate Server enabled.
%PKI-6-CS_ENABLED: Certificate server now enabled.

// Export the root PKI certificate of the Root-CA device to the flash directory
IOS-Root-CA(config)# crypto pki export PKI pem url flash:
% The specified trustpoint is not enrolled (PKI).
% Only export the CA certificate in PEM format.
% Exporting CA certificate...
Destination filename [PKI.ca]?
Writing file to flash:PKI.ca

// Share the PKI certificate from the Root_CA device via TFTP so that it can be
downloaded by other devices
IOS-Root-CA(config)# tftp-server flash:PKI.ca
```

Briefly, the Root certificate that has been created will be used on each controller and WAN-edge device to establish 'trust'. The image below shows the form of the root certificate from the Root_CA device that has been created.



```

Terminal
Root_CA #
IOS-Root-CA(config)#crypto pki export PKI pem terminal
% The specified trustpoint is not enrolled (PKI).
% Only export the CA certificate in PEM format.
% CA certificate:
-----BEGIN CERTIFICATE-----
MIIDFDCCAfygAwIBAgIBATANBgkqhkiG9w0BAQsFADAbMRkwFwYDVQQDExByb290
Y2EubGfiLmxvY2FsMB4XDTI0MTEzMDA4MjAwNv0XTI3MTEzMDA4MjAwNv0GzEZ
MBcGA1UEAxMQcm9vdGNhLmxhYi5sb2NhDCCASIwDQYJKoZIhvcNAQEBBQADggEP
ADCCAQoCggEBAMSDQ+NewY4N8trdfnT+c765r3DCwRmX9Lr5FK9LjDnygdzVE6w4
nmwLkWRosMhTv4XfRRwPdT8mMotR0BzB5qrs7IRW+gcC3vUMmA/tynXGnFrSg3ol
WDILcDriFo+0AsyqkdVrTa8expPQmMe8GzBsdE03rMxW+XnVFYNn6V89yYbFVN10
d1+nERkomglny18Uwa4xDPxEJC9wMI8/0FzoDGwtw2LpSTiEMUWIWq4aHCzjXwzX
730eYuTjVMhE5X2m+J6KAdvaMkjRjdMAY0wjjBoov1vjh6Hp8Ev0ub0Gy0jd3f
JKSrGipUx+NQ9Buvp07daR8zMUPeBfvPLY0CAwEAaNjMGewDwYDVR0TAOH/BAUw
AwEB/zAOBgNVHQ8BAf8EBAMCAYYwHwYDVR0jBBgwFoAUoANMYAyIyS6QA5IeV72k
PBw0CV8wHQYDVR00BYEFKADEGAMiMkuKAOSHLe9pDwcNALfMA0GCSqGSIb3DQE8
CwUA4IBAQEcVcnx70ddAFaaauHytaPzEWsPWGiH49G2snJlqpOws2ZHkw0Fg7
uixFn9GGWE6Pz6zYpg0CTV/vHekWMudXwpF7AFID/KLpT3tnUSPEFjs1s1yxDG
2xemF3vseu2WCQmw0rynr9EWQ5a/hTfrbMiLnn2wcmcMBuSlBhjyW1Ue7cWCAPu
Xp5lDvfQ90kw+AkEvR/rsTVjD5Ht1yqeZtap7Ac2Wmr/ZxtRU0Aok+VfQP2lL1k6
0PzNN81fx0w2PSV2ncZ47tmBsGCOVBr+b9lkm6DXLEGv0UC9238lLbHwsvoWVyw2
3KESDx0h/hioKA1l0G7/TtQIqXq5NdaB
-----END CERTIFICATE-----

```

2. Setup vManage Certificate: download & install root-ca cert, generate CSR

*notes: Debug the Root_CA device to check whether the controller is initiating a TFTP request or not. If no debug output appears, use the vshell option!

```
Root_CA# debug tftp events/packets
```

// Download the root certificate via TFTP from the Root_CA device

```
vManage# request download tftp://100.10.1.100/PKI.ca
```

----- jika tidak bisa, download lewat vshell -----

```
vManage# vshell
```

```
vManage:~$ tftp -g -r PKI.ca 100.10.1.100
```

// Install the root certificate downloaded from the Root_CA on the controller.

```
vManage# request root-cert-chain install home/admin/PKI.ca
```

// Establish trust between vManage and Root_CA, import the root certificate into vManage, and prepare the CSR (Certificate Signing Request)

1. Log in to the vManage GUI.
2. In the side menu, go to Administration > Settings.

3. Edit the Organization name.
4. Specify the vBond.
5. Change 'Controller Certificate Authorization' to 'Enterprise Root Certificate'.

Controller Certificate Authorization	Manual
<input type="radio"/> Symantec Automated (Recommended) <input type="radio"/> Symantec Manual <input checked="" type="radio"/> Enterprise Root Certificate	

6. Manually copy the root certificate from the Root_CA device (you can use the command '**crypto pki export PKI pem terminal**')

```

IOS-Root-CA(config)#crypto pki export PKI pem terminal
% The specified trustpoint is not enrolled (PKI).
% Only export the CA certificate in PEM format.
% CA certificate:
-----BEGIN CERTIFICATE-----
MIIDFDCCAfygAwIBAgIBATANBgkqhkiG9w0BAQsFADAbMRkwFwYDVQQDExByb290
Y2EubGFilmxvY2FsMB4XDTI0MTezMDA4MjAwNv0xDTl3MTezMDA4MjAwNvowGzEZ
MBcGA1UEAxMQcm9vdGhLmxhYi5sb2NhDCCASiwDQYJKoZIhvcNAQEBBQADggEP
ADCCAQoCggEBAMSD0+NeWY4N8trdfnT+c765r3DCwRmX9Lr5FK9LjDnygdzVE6w4
nmwLkWRosMhTv4XfRRwPdT8mMotR0bZB5qrs7IRW+gcC3vUMmA/tynXGnFrSq3o1
WDILcDriFo+0AsyqkdVrTa8expP0mMe8GzBsdeQ3rMxW+XnVFYNN6V89yYbFVN10
d1+nERkomglny18Uwa4xDpxEJC9wMI8/0FzoDGwtw2LpSTiEMUWIwq4aHCzjXwZX
730eYuTjVMMhE5X2m+J6KAdvaMkjRjdMAY0wjBoov1Vjh6Hp8Ev0ub0Gy0jd3f
JKSrGipUx+NQ9Buvp07daR8zMuPeBfvPLY0CAwEAaAjNjMGewDwYDVR0TAQH/BAUw
AwEB/zAOBgNVHQ8BAf8EBAMCAYyHwYDVR0jBBgwFoAUoANMYAyIyS6QA5IeV72k
PBw0CV8wHQYDVR0OBByEFKADTGMiMkuAOSHle9pDwCNAlfMA0GCSqGSIb3DQE
CwUAA4IBAQDEcVcnx70ddAFaauuHYtaPzEWspWGiH49G2snJlqp0ws2ZHkw0Fg7
iuiXfn9GGWE6Pz6zYpg0CTV/vHekWMudXwpF7AFID/KLpT3tnU5PEFjS1s1yxDG
2xemF3vseu2WCQmw0ryN9EWQ5a/hTfrbMln2wcmczbUsBlhjyW1Ue7cWCAPu
Xp5ldvf090kw+AkEvR/rsTVjD5HtlyqeZtap7Ac2Wmr/ZxtRUOAok+VfQP2LL1k6
OZpZN8ifX0w2P5V2ncZ47tmBsGCOVBr+b9lkm6DXLEGv0UC92381LbHwsV0WVw2
3kESDx0h/h0kA1l0G7/TtQIqXq5NdaB
-----END CERTIFICATE-----

```

7. Paste the root certificate that has been copied into 'Controller Certificate Authorization'

Controller Certificate Authorization	Cisco
<input type="radio"/> Cisco Automated (Recommended) <input type="radio"/> Symantec Automated <input type="radio"/> Manual <input checked="" type="radio"/> Enterprise Root Certificate	
Certificate <pre> -----BEGIN CERTIFICATE----- MIIDFDCCAfygAwIBAgIBATANBgkqhkiG9w0BAQsFADAbMRkwFwYDVQQDExByb290 Y2EubGFilmxvY2FsMB4XDTI0MTezMDA4MjAwNv0xDTl3MTezMDA4MjAwNvowGzEZ MBcGA1UEAxMQcm9vdGhLmxhYi5sb2NhDCCASiwDQYJKoZIhvcNAQEBBQADggEP ADCCAQoCggEBAMSD0+NeWY4N8trdfnT+c765r3DCwRmX9Lr5FK9LjDnygdzVE6w4 nmwLkWRosMhTv4XfRRwPdT8mMotR0bZB5qrs7IRW+gcC3vUMmA/tynXGnFrSq3o1 d1+nERkomglny18Uwa4xDpxEJC9wMI8/0FzoDGwtw2LpSTiEMUWIwq4aHCzjXwZX 730eYuTjVMMhE5X2m+J6KAdvaMkjRjdMAY0wjBoov1Vjh6Hp8Ev0ub0Gy0jd3f JKSrGipUx+NQ9Buvp07daR8zMuPeBfvPLY0CAwEAaAjNjMGewDwYDVR0TAQH/BAUw AwEB/zAOBgNVHQ8BAf8EBAMCAYyHwYDVR0jBBgwFoAUoANMYAyIyS6QA5IeV72k PBw0CV8wHQYDVR0OBByEFKADTGMiMkuAOSHle9pDwCNAlfMA0GCSqGSIb3DQE CwUAA4IBAQDEcVcnx70ddAFaauuHYtaPzEWspWGiH49G2snJlqp0ws2ZHkw0Fg7 iuiXfn9GGWE6Pz6zYpg0CTV/vHekWMudXwpF7AFID/KLpT3tnU5PEFjS1s1yxDG 2xemF3vseu2WCQmw0ryN9EWQ5a/hTfrbMln2wcmczbUsBlhjyW1Ue7cWCAPu Xp5ldvf090kw+AkEvR/rsTVjD5HtlyqeZtap7Ac2Wmr/ZxtRUOAok+VfQP2LL1k6 OZpZN8ifX0w2P5V2ncZ47tmBsGCOVBr+b9lkm6DXLEGv0UC92381LbHwsV0WVw2 3kESDx0h/h0kA1l0G7/TtQIqXq5NdaB -----END CERTIFICATE----- </pre>	
<input type="checkbox"/> Set CSR Properties	

8. Check 'Set CSR Properties', then fill in the identity certificate, import, and save

Set CSR Properties

Domain Name
mrc.org

Organizational Unit
MRC

Organization
MRC

City
Jakarta

State
Jakarta

Email
ridhoc@mrc.org

2-Letter Country Code
ID

Validity
1 Year

Import & Save **Cancel**

9. Thus, the CSR has been successfully created and trust has been established between vManage and Root CA.

Controller Certificate Authorization	Enterprise	View Edit
WAN Edge Cloud Certificate Authorization	Automated	View Edit
Web Server Certificate	12 Jul 2024 11:30:30 PM	CSR Certificate

// Generate the CSR, then request the Root_CA to sign the CSR

1. Go to the vManage GUI menu, Configuration > Certificate > Controllers tab

CONFIGURATION | CERTIFICATES

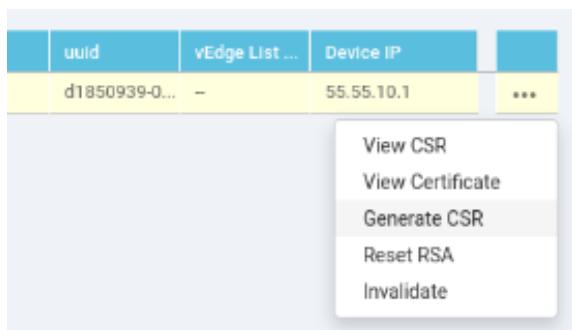
WAN Edge List **Controllers**

Send to vBond

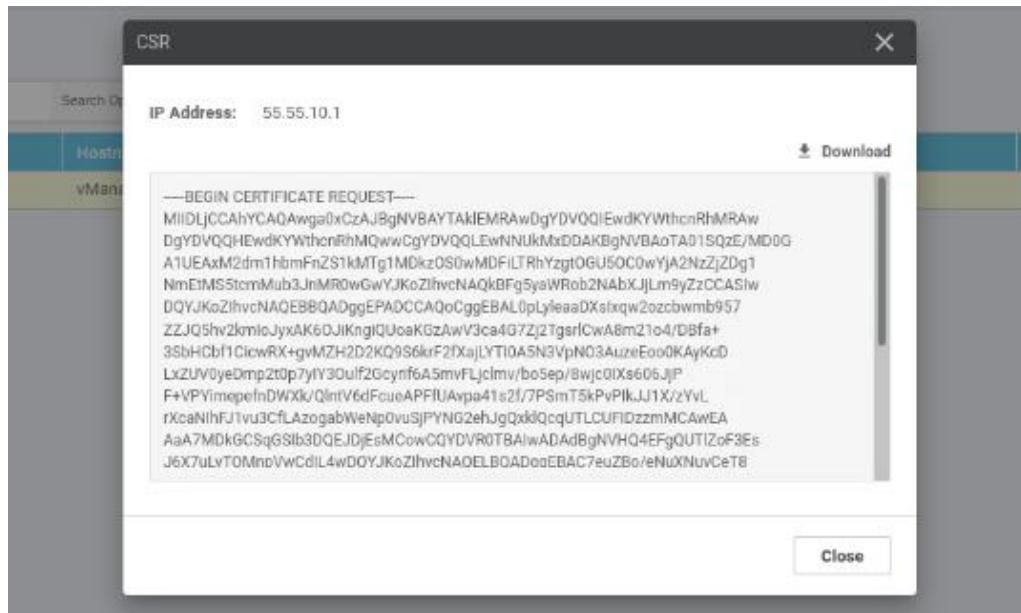
Total Rows: 1

	Operation Status	Controller Type	Hostname	System IP	Site ID	Certificate Serial	Expiration Date	uuid	vEdge List ...	Device IP	...
>	CSR Generated	vManage	vManage	55.55.10.1	10	No certificate installed	—	d1850939-0...	—	55.55.10.1	***

2. In the far-right menu (...), select 'Generate CSR'



3. Copy the certificate/CSR.



4. Go to the Root_CA device, then execute the following configuration:

IOS-Root-CA# crypto pki server PKI request pkcs10 terminal

*This is intended to submit a digital certificate request to the Certificate Authority (CA)

**IOS-Root-CA#crypto pki server PKI request pkcs10 terminal
PKCS10 request in base64 or pem**

% Enter Base64 encoded or PEM formatted PKCS10 enrollment request.
% End with a blank line or "quit" on a line by itself.

5. Paste the CSR that was copied from vManage earlier.

```

% Enter Base64 encoded or PEM formatted PKCS10 enrollment request.
% End with a blank line or "quit" on a line by itself.
-----BEGIN CERTIFICATE REQUEST-----
MIIDLjCCAHYCAQAwga0xCzAJBgNVBAYTAKIEMRAwDgYDVQQIEwdKYWthcnRhMRAw
DgYDVQQHEwdKYWthcnRnQwwCgYDVQQLEwNNUkMxDDAKBgNVBAoTA01SQZE/MDBG
A1UEAxM2dm1hbmfNzS1kMTg1MDkzOS0wMDF1LTrhYzgtOGU5OC0wYJA2NzzjZDg1
NmEtMS5tcmMub3JnMR0wGwYJKoZlhwvNAQkBFG5yaWRob2NAbXJjLm9yZzCCASiw
DQYJKoZlhwvNAQEBBQADggEPADCCAQoCggEBAL0pLyIeaaDXsIxqw2ozcbwmb957
ZZJQ5hv2kmloJyxAK60JIKnglQJUoaKGzAwV3ca4G7zj2TgsrlCwA8m21o4/DBfa+
35bHCbf1CicwRX+gvMZH2D2KQ956krF2fxajLYTI0ASN3VpNO3AuzeEoo0KAyKcD
LxZUV0yeDmp2t0p7yIY30ulf2Gcyrif6A5mvfLjclmv/bo5ep/BwjcoIXs606JJp
F+VPYimepefnDWXk/QlntV6dFcueAPFFUAvpa41s2f/7PSmT5kPvPIkJJ1X/zYvL
rXcaNiHfJ1vu3CFLAzogabWeNp0vuSjPYNG2ehJgQxklQcqUTLCUFIDzzmMCAwEA
AaA7MDkGCSqGSib3DQEJDjEsMCowCQYDVR0TBAlwADAdBgNVHQ4EFgQUTlZoF3Es
J6X7uLVTOmnpVwCdIL4wDQYJKoZlhwvNAQELBQA0DggEBAC7euZBo/eNUXNuvCeT8
2y7kIKSuNartUCL2z28R0yZNwY5F1JY2EbFbubd/LNbNXzzBAsYM28DI8YJvESSUg
eQdMLXKOjfHClk689+ToYDXCdPqOxp5At7YNY5TnqvL4rMWhbr90tlJN6nq4vSc1
bKroi2l0qMm54uTonfM/lptdELv+dnQp27AldtJ2pif3boGOSSfq8j0ZowIEX1TR
PILHe4Ee+6YY5YSNXAXwnPzULA2DYL4LQx/sPyBapf4P9kq7yBevX/MTu8YtuTg2
J1qheekFZPgeU64qFrFyFHEoUkrQbpgpdiu7HfYtrE5ScZFAaYcqPcBZYwwoEKT7
QxE=
-----END CERTIFICATE REQUEST-----

```

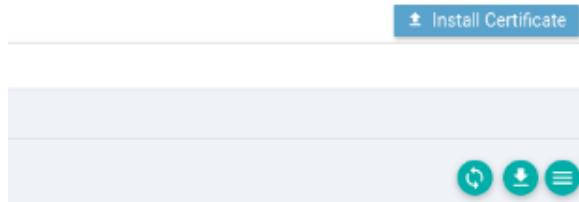
6. After that, type 'quit' and the signed certificate will be issued (granted!)

```

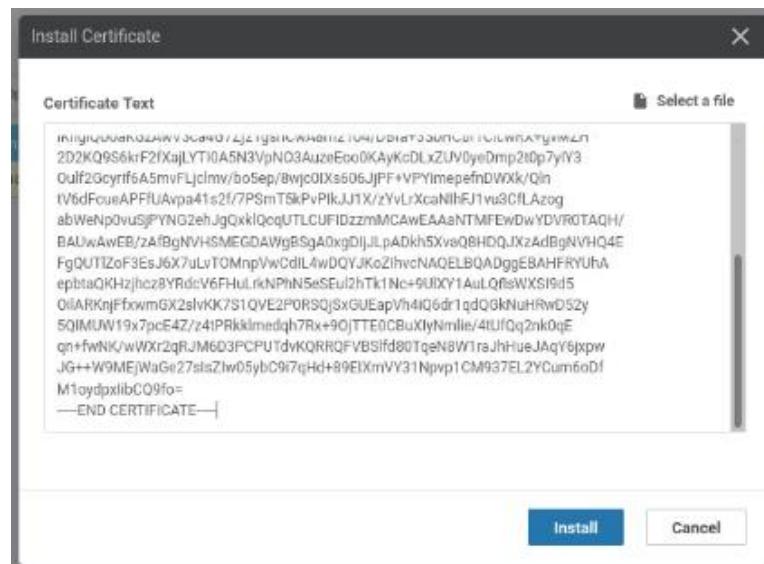
quit
% Granted certificate:
-----BEGIN CERTIFICATE-----
MIIDlzCCAn+gAwIBAgIBAjANBgkqhkiG9w0BAQsFADAbMRkwFwYDVQQDExByb290
Y2EubGF1LmxvY2FsMB4XDTI0MTEzMDExMxN1oXTDI1MTEzMDEzMzExMxN1owga0x
CzAJBgNVBAYTAKLEMRAwDgYDVQQIEwdKYWthcnRhMRAwDgYDVQQHEwdKYWthcnRh
MQwwCgYDVQQLEwNNUkMxDAAKBgNVBAoTA01SQzE/MDBGA1UEAxM2dm1hbFnZS1k
MTg1MDkzOS0wMDFlLTRhYzgtOGU5OC0wYja2NzzjZDg1NmEtMS5tcMUb3JnMR0w
GwYJKoZIhvcNAQkBFG5yaWRob2NAhXjjLm9yZzCCASIwDQYJKoZIhvcNAQEBBQAD
ggEPADCCAQoCggEBAL0pLyleaaDXsIxqw2oZcbwmb957ZZJQ5hv2kmIoJyxAK6OJ
lKngtQUoaKGzAwV3ca4G7zj2TgsrlCwA8m21o4/DBUa+3SbHCbf1CicwRX+gvMZH
2D2KQ9S6krF2fxajLYT10A5N3VpNO3AuzeEoo@KAyKcDLxZUV@yeDmp2t@p7yIY3
0ulf2Gcyrif6A5mvFLjclmv/b0Sep/Bwjc0IXs606JJPF+VPYimepefnDWXk/Qln
tv6dFcueAPFfUAvgpa41s2f/7PSmT5kPvPIkJJ1X/zYvLrXcaNlhFJ1vu3CfLAzog
abWeNp0vusJPYNG2ehJgQxklQcqUTLCUFIDzzmMCawEAAAATMFEwDwYDVROTAQH/
BAUwAwEB/zAFBgNVHSMEGDAwB5gA0xgDiJLpADkh5XvaQ8HDQJXzAdBgNVHQ4E
FgQUTlZoF3EsJ6X7uLvTOMnpVwCdIL4wDQYJKoZIhvcNAQELBQADggEBAHF瑞UhA
epbtaQKHjhcZBYRdcVGFHuLrkNphN5eSEu1hTk1Nc+9UlXY1AuLQflsWXSI9d5
01lARKnjFfxwmGX2slvKK71QE2P0RSQjSxGUeapvh4iQ6dr1qdQGKnUHRwD52y
5QIMUW19x7pcE4Z/z4tPRkkImedqh7Rx+90JTT0CBuXiyNmllie/4tUfQq2nk0qE
qn+fwnK/wlxr2qRJM6D3CPUTdVVKQRQFVB5lf80TqeN8W1raJhHueJAqY6jxpw
JG++W9MEjWaGe27sIsZIw05ybC9i7qHd+B9EIXmVY31Npvip1CM937EL2YCum6oDf
M1oydpXlibCQ9fo=
-----END CERTIFICATE-----

```

7. Copy the granted certificate, then return to the vManage GUI. In the top-right corner, click 'Install Certificate'



8. Finally, paste the granted certificate into the 'Install Certificate' field, then click install.



9. Wait for a moment, and the certificate will be successfully installed on vManage.

The screenshot shows the 'TASK VIEW' section of the Cisco vManage interface. It displays a single task named 'Install Certificate' with a status of 'Success'. The log details the process: it started processing a serial list file, completed processing, pushed the vSmart List, pushed the serial list to vManage, and finally successfully synced the vEdge list. The device ID is d1850939-001b-4ac8-8e98-0b0676cd856a and the system IP is 55.55.10.1.

10. If we go back to Configuration > Certificate, we can see that the certificate has been successfully installed on vManage

The screenshot shows the 'CONFIGURATION | CERTIFICATES' page under the 'Controllers' tab. It lists a single certificate entry for 'vManage' with the hostname 'vManage', system IP '55.55.10.1', and expiration date '30 Nov 2025 1:33:17 PM GMT'. The status is 'vBond Updated'. The log shows three events: CSR Generated, Installed, and vBond Updated.

3. Setup vBond Certificate: download & install root-ca cert, generate CSR

// Add the vBond device to vManage

1. Go to the vManage GUI, Configuration > Device > Controller, then click 'Add Controller' and select vBond.

The screenshot shows the 'CONFIGURATION | DEVICES' page under the 'Controllers' tab. A new controller named 'vBond' has been added, listed alongside 'vSmart'. The table shows the controller type, hostname, system IP, site ID, mode, and assignment status. The vBond entry has 'vManage' as the hostname, '55.55.10.1' as the system IP, '10' as the site ID, 'CLI' as the mode, and '-' as the assignment status.

2. Enter the vBond IP address along with its username and password, then check 'Generate CSR'.

Add vBond

vBond Management IP Address
100.10.1.12

Username
admin

Password

Generate CSR

Add Cancel

3. Thus, the vBond device has been added to vManage

Controller Type	Hostname	System IP	Site ID	Mode	Assigned Template	Device Status	Certificate Stat...	Pol...
vManage	vManage	55.55.10.1	10	CLI	--	In Sync	Installed	--
vBond	-	-	--	CLI	--		Not-Installed	--

// Download the root certificate via TFTP from the Root_CA device

```
vBond# request download tftp://100.10.1.100/PKI.ca
```

----- If it is not possible, download via vshell -----

```
vBond# vshell
```

```
vBond:~$ tftp -g -r PKI.ca 100.10.1.100
```

// Install the root certificate downloaded from the Root_CA on the controller

```
vBond# request root-cert-chain install home/admin/PKI.ca
```

```
// Generate the CSR, then request the Root_CA to sign the CSR
```

1. Go to Configuration > Certificate, then under vBond, you can see 'CSR Generated'. Select 'View CSR'

uuid	Operation Status	Site ID	Certificate Serial	vEdge List ...	Device IP	...
e7d9c4cb-4...	CSR Generated	-	No certificate installed	Sync	100.10.1.12	***
d58269df-8...	vBond Updated	10	02	-	5	<div style="border: 1px solid #ccc; padding: 5px; display: inline-block;"> View CSR View Certificate Generate CSR Reset RSA Invalidate </div>

2. Copy the CSR from the vBond.



3. Return to the Root_CA device, then use the command 'crypto pki server PKI request pkcs10 pem terminal', and paste the CSR from the vBond that was copied.

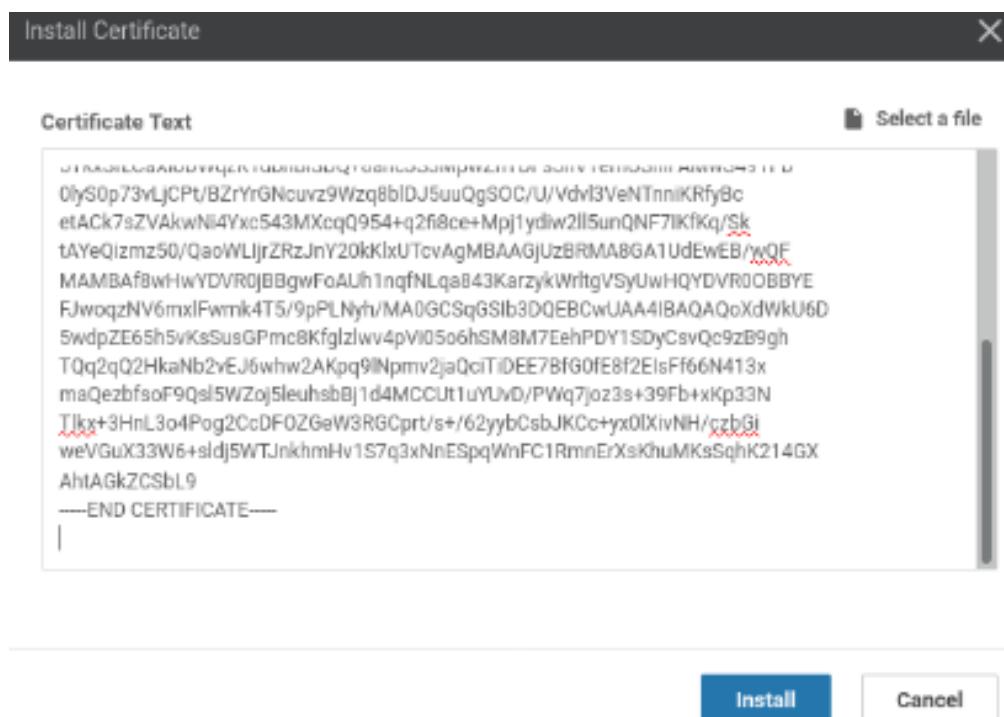
```
IOS-Root-CA#crypto pki server PKI request pkcs10 terminal
PKCS10 request in base64 or pem

% Enter Base64 encoded or PEM formatted PKCS10 enrollment request.
% End with a blank line or "quit" on a line by itself.
-----BEGIN CERTIFICATE REQUEST-----
MIIDLDCCAhhQCAQAwgasxCzAJBgNVBAYTAKleEMRAwDgYDVQQIEwdKYWthcnRhMRaw
DgYDVQHewdKYNthcnRhNQwwCgYDVQQLewNNUKMxDDAKBgNVBAoTA015QzE9HDsg
A1UEAxM0dnJvbmc0Tdk0WHM0YzgTNDRLNS00OWyJWLIxYNMty2U2ZGzk0WUSNHE4
LTaUbXjJLm9yZzEdMBsGCSqSIB3DQEJARYOcmIkaG9jQG1yYy5vcmcwggiMA0G
CSqGSIb3DQEBAQUA041BdAwggEKAoIBAQc/kh5T2LUa9hy6s8WORwN928Tgxg
700+XuK9JrML3woL3wEmodys/z9pydfuUapYpQ89IUfmofnkTzxAVqZ64YhfzzCN
zy34IHukr2dx/Jyl-m3e5YkxSflCaXloDWqzK1dbnBLSDQ+8ahcJS3MPwZhTDFs5
nV1ernuShLFAMwS49TFD0lyS0p73vLjCpt/BZrYrGNcvz9Wzq8b1D35uuQgSOC/
U/Vdv13VeNTnniKRyBcetAck7sZVAkwNi4Yxc543NXcq0954+q2fi8ce+Mpj1yd
iw2llsunQNFT1KfKo/SktAYeQizmz8/QaoWLijrZrJny28kkLxUTcvAgMBAAGg
OzASBgkqhkiG9w0BCQ4XLDAgMAKGA1UDewQCMAAWHQYDV00BBYEFJwoqzNV6mxl
Fwmk4T5/9pPLNyh/MA0GCSqSIB3DQEBCwUA4IBAQB+dglEs+HsJ3VyzDefxlc
Es1se8jeGdNwD3QJD/pry7h0f/XMEgBN/pteLSRlWjKwj01wbvSLVagHeYeZMFk0
8xgdrulqoatSPImB0dTNRmpfPbAfq0m9h6t2vaFKHMuImbeN4w2jaxLoF9zf
+NTeENPe+tHE3be2Jii/a8WzX8d19HNKCVWabzVHoD8ei7AVmVscI6U320w6/Rn
lx3br8e5yh70PK4XDZtPDvKm7j8fq610IUoZ1x3D500kAfbcIncv0b8GM9qfu/m+
pV2xtgxKK115KyrzQjHYe03Eo0W96hzHvv7AkPhy5fH8YQbS75gv5vYXVKazs758
-----END CERTIFICATE REQUEST-----
```

4. Type 'quit', then the granted certificate will be issued by the Root_CA. Copy the granted certificate.

```
-----END CERTIFICATE REQUEST-----
quit
% Granted certificate:
-----BEGIN CERTIFICATE-----
MIIDLTCAn2gAwIBAgIBAzANBgkqhktG9w0BAQsFADAbMRkwFwYDVQQDEXByb290
Y2EubGFiLmxvY2FsMB4XDTI0MTIwMTEyNTQ0NFoXDTI1MTIwMTEyNTQ0NFowgasx
CzAJBgNVBAYTAkLEMRAwDgYDVQQIEwdKYWthcnRhMRAwDgYDVQQHEwdKYWthcnRh
MQwwCgYDVQQLEwNNUkMxDDAKBgNVBAoTA01SQzE9MDsGA1UEAxM0dmJvbmtZTdk
0WM0YzgtNDRlNS00OWVjLWIXYWMtY2U2ZGzk0NU5NWE4LTaubXjjLm9yZzEdMBsG
CSqGSIB3DQEJARYOcmIkaG9jQG1yYy5vcncwgEtMA0GCSqGSIB3DQEBAQUAA4IB
DwAwggEKAoIBAQc/kH0/2TYLuA9hY6s8WQRm928Tgxg7o0+XuK9JrML3woL3wEm
odys/zpydfrUapYpQ89IUFmOfnkTzxAVqZ64YhFzzCNzy34IHukr2dx/JyL+m3e
5VkxSfLCaXloK1dbnBiSDQ+BahcJS3MpwhTDFs5v1ernUShlFAMwS49TFD
0lyS0p73vLjCPT/BZrYrGNcuvz9Wzq8blDJ5uuQgSOC/U/Vdvl3VeNTnniKRfyBc
etACK7sZVAkwNi4Yxc543MXcqQ954+q2f18ce+Mpj1ydiw2ll5unQNF7IKfKq/Sk
tAYeQizmz50/QaoWLJjrZrJnY20kKlxUTcvAgMBAAGjUzBRMA8GA1UdEwEB/wQF
MAMBAf8wHwYDVR0jBBgwFoAUh1nqFNlqa843KarzykWrItgVSyUwHQYDVR0OBByE
FJwoqzNV6mxlfwmk4T5/9pPLNyh/MA0GCSqGSIB3DQEBCwUAA4IBAQAAQoXdwkU6D
5wdpZE65h5vKssusGPmc8Kfglzlwv4pVi05o6hSM8M7EehPDY1SDyCsvQc9zB9gh
TQq2qQ2HkaNb2vEJ6whw2AKpq9lNpmv2jaQc1TlDEE7BfG0fE8f2EIsFf66N413x
maQezbfs0F90sl5WZoj5leuhsb8j1d4MCCU1uYUvD/PWq7jaz3s+39Fb+xKp33N
T1kx+3HnL3o4Pog2CcDF0ZGeW3RGCprt/s+/62yybCsbJKCc+yx0lXivNH/czbGi
weVGuX33W6+sldj5WTJnkhmHv157q3xNnESpqWnFC1RmnErXsKhuMKsSqhK214GX
AhtAGkZCSbL9
-----END CERTIFICATE-----
```

5. Go back to the vManage GUI, then in the top-right corner, click 'Install Certificate', paste the granted certificate that was copied, and then click 'Install'



6. Thus, the vBond device has been successfully installed on vManage or successfully onboarded

The screenshot shows the vManage Configuration > Devices page. At the top, there is a log entry:

```
[1-Dec-2024 19:58:02 WIB] Completed processing serial list file on vBond-e7d9c4c8-44e5-49ec-b1ac-ce6df9e95a8
[1-Dec-2024 19:58:02 WIB] Done - Push vSmart List for vBond-e7d9c4c8-44e5-49ec-b1ac-ce6df9e95a8
[1-Dec-2024 19:58:02 WIB] Pushed serial list to vSmart list on vBond-e7d9c4c8-44e5-49ec-b1ac-ce6df9e95a8 ()
[1-Dec-2024 19:58:02 WIB] Successfully synced vSmart list on vBond-e7d9c4c8-44e5-49ec-b1ac-ce6df9e95a8
[1-Dec-2024 19:58:02 WIB] No new updates to be sent to device
[1-Dec-2024 19:58:03 WIB] Successfully synced vEdge list on vBond-e7d9c4c8-44e5-49ec-b1ac-ce6df9e95a8
```

Below the log, the 'Controllers' tab is selected. It shows the following table:

Controller Type	Hostname	System IP	Site ID	Mode	Assigned Template	Device Status	Certificate Stat...
vManage	vManage	55.55.10.1	10	CLI	--	In Sync	Installed
vBond	vBond	55.55.10.2	10	CLI	--	In Sync	Installed

4. Setup vSmart Certificate: download & install root-ca cert, generate CSR

// Add the vSmart1 device to vManage

1. Go to the vManage GUI, Configuration > Device > Controller, then click 'Add Controller' and select vSmart.

The screenshot shows the vManage Configuration > Devices page. The 'Controllers' tab is selected. In the 'Add Controller' dropdown, 'vSmart' is highlighted. The table below shows the current controllers:

Controller Type	Hostname	System IP	Site ID	Mode	Assigned Temp
vManage	vManage	55.55.10.1	10	CLI	--
vBond	vBond	55.55.10.2	10	CLI	--

2. Enter the vSmart1 IP address along with its username and password, then check 'Generate CSR'. Leave the protocol and port as default.

Add vSmart

vSmart Management IP Address
100.10.1.13

Username
admin

Password

Protocol
DTLS

Port

Generate CSR

Add Cancel

3. Thus, the vSmart device has been added to vManage.

WAN Edge List Controllers

Add Controller Change Mode

Controller Type	Hostname	System IP	Site ID	Mode	Assigned Template	Device Status	Certificate Stat.
vManage	vManage	55.55.10.1	10	CLI	--	In Sync	Installed
vSmart	--	--	--	CLI	--		Not-Installed
vBond	vBond	55.55.10.2	10	CLI	--	In Sync	Installed

// Download the root certificate via TFTP from the Root_CA device

vSmart1# request download tftp://100.10.1.100/PKI.ca

----- If it is not possible, download via vshell -----

vSmart1# vshell

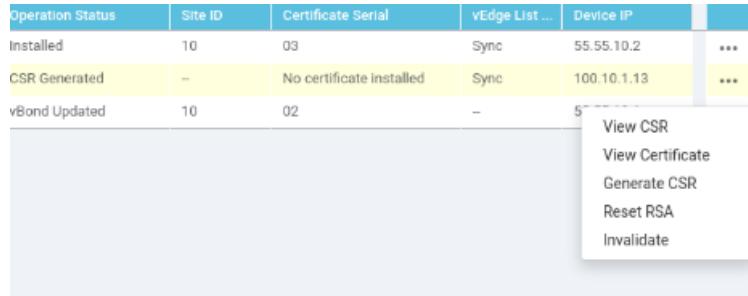
vSmart1:~\$ tftp -g -r PKI.ca 100.10.1.100

// Install the root certificate downloaded from the Root_CA on the controller.

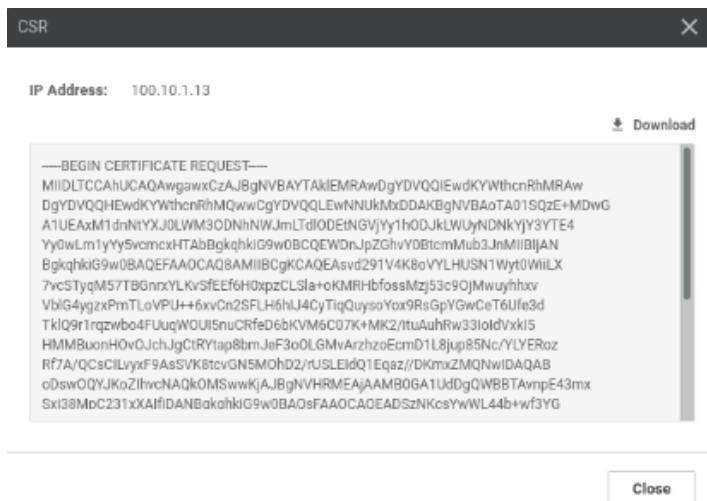
vSmart1# request root-cert-chain install home/admin/PKI.ca

// Generate the CSR, then request the Root_CA to sign the CSR

1. Go to Configuration > Certificate, then under vSmart1, you can see 'CSR Generated'. Select 'View CSR'



2. Copy the CSR from vSmart1.



3. Return to the Root_CA device, then execute the command 'crypto pki server PKI request pkcs10 pem terminal', and paste the CSR from vSmart1 that was copied.

```
IOS-Root-CA#crypto pki server PKI request pkcs10 terminal
PKCS10 request in base64 or pem

% Enter Base64 encoded or PEM formatted PKCS10 enrollment request.
% End with a blank line or "quit" on a line by itself.
-----BEGIN CERTIFICATE REQUEST-----
MIIDLTCAhUCAQAwgawxCzAJBgNVBAYTAlEMRAwDgYDVQQIewdKYWthcnRhMRAw
DgYDVQQHEwdKYWthcnRhMQRwwCgYDVQQLEwNNUKmxDDAKBgNVBAoTA01S0Zc+MDwG
A1UEAxM1dnNtYXJ0LWM3ODNhNjM1Tld0DETNGVjYy1hODjkLWUyNDNkYjY3TE4
Yy0wLm1yy5vcmcxITAbBgkqhkiG9w0BCQEWDnJpZGhV0BtcmMu3JnMIIBIJAN
BgkqhkiG9w0BAQEFAAOCAQ8AMIIIBCgKCAQEAsvd291V4K8oVYLHUSN1wyt0WiILX
7vcStyqM57TBGnrxyLKvsfEEf6h0xpzCLSlr+oKMRHbfossMzj53c90|Mwuyhhxv
VbLG4ygzxPmTLoVPU++6xVCnZ5FLH6h1J4cyTiqQuys0yo9RsGpYGwCeT6ufe3d
TklQ9r1rqzb04FuugWOUISnuCrfe6bKVh6C07K+MK2/ltuAuRw33IoIdVxk15
HMMBuonH0v0JchJgCtRYlap&bmJef3oOLGwArzhzoEcmD18jup85NC/YLERoz
RF7A/QCsCILvyxF9asSVK8tcvGN5M0hD2/rUSLEldQ1Eqaz//DKmxZMQNwIDAQAB
oDsw0QYJKoZIhvNAQkOMSwwKjAJBgNVHRMEAjAAMB0GA1UdDgQWBBAvnpE43mx
Sxi38MpC231xXAIfjDANBgkqhkiG9w0BAQsFAAOCAQEADESzNkcsYwHL44b+wf3YG
EL8oAqEPTRpHruk3th0QisP9g28tl3oB4JZJHq9+Nv+98cgsoZlaZsWI5oxak
VsbcGldN7d/4+54ATM1kJ5gj/aTHnBtlKheMskeZbRvC3al1WhzBPY2dzXfp6
GqYodVT3Q4o+CbkB3trg@T6VymEZNNphGLC+qwMpo60/PoTgM58ANOT0NB3koxem
d/UMUQpjxbvcFVi+a60Fyl918gbmlmZq5ngcbvYkj6LG61oDsugs8ZQlH3KJ11
I/621h5LDgi0dAEHF9EcBYa/P6n//Llo/jjjg3KVP3m7Rmz1dtmBB0SmSzqyrrc
WQ==
-----END CERTIFICATE REQUEST-----
```

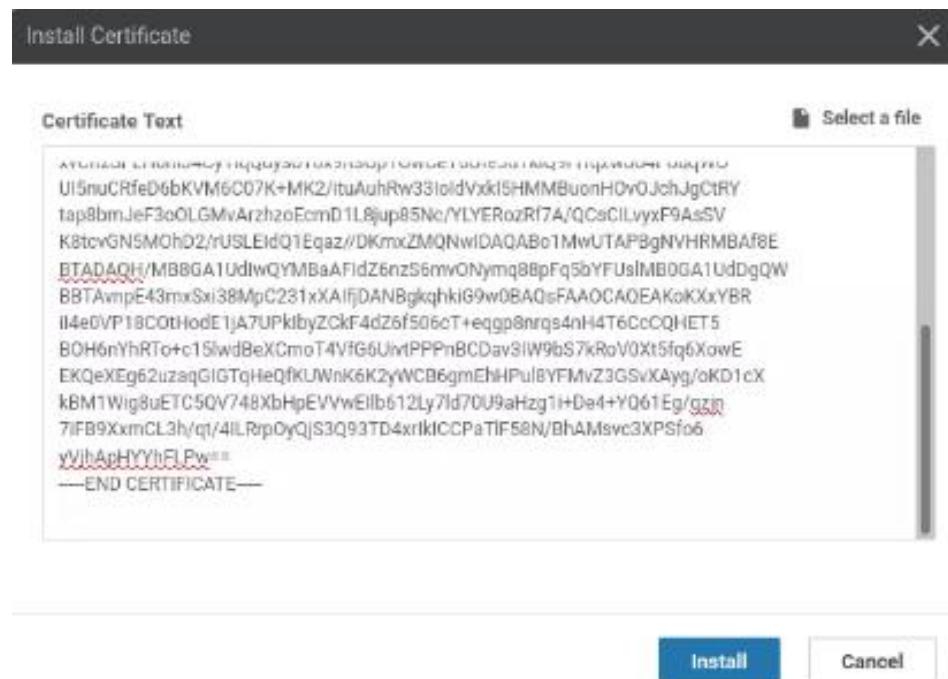
4. Type 'quit', then the granted certificate will be issued by the Root_CA. Copy the granted certificate.

```

quit
% Granted certificate:
-----BEGIN CERTIFICATE-----
MIIDLjCCAn6gAwIBAgIBBDANBgkqhktG9w0BAQsFADAbMRkwFwYDVQQDExByb290
Y2EubGF1LmxvY2FsMB4XDTI0MTIwMTEzNTAzMloXDTI1MTIwMTEzNTAzMlowgawx
CzAJBgNVBAYTAkLEMRAwDgYDVQQIEwdKYWthcnRhMRAwDgYDVQQHEwdKYWthcnRh
MQwwCgYDVQQLEwNNUkMxDDAKBgNVBAoTA01SQzE+MDwGA1UEAxM1dnNtYXJ0LWM3
ODNhNWJmLTdLODEtNGVjYy1hODJkLWUyNDNkYjY3YTE4Yy0wLm1yYy5vcmcxHTAb
BgkqhktG9w0BCQEWdNjpZGhvY0BtcMUb3JnMIIBiJANBgkqhktG9w0BAQEFAAOc
AQ8AMIIIBCgKCAQEAsvd291V4K8oVYLHUSN1Wyt0WtIiLX7vcSTyqM57TBGnrxFYLkv
SFEEF6H0xpzCLSLa+oKMRRhbfoSSMzj53c90jMwuyhhxvBbLG4ygzxPmTLoVPU++6
xvCn2SFLH6hIJ4CyTiqQuys0Yox9RsGpYGwCeT6Ufe3dTklQ9r1rqzwbo4FUuqW0
UI5nuCRfeD6bKVM6C07K+MK2/ituAuhRw33IoIdVxk1SHMMBuonH0v0JchJgCtRY
tap8bmJeF3oOLGMvArzhzoEcMD1L8jup85Nc/YLYERozRf7A/QCsCILvyxF9AsSV
K8tcvGN5MOhD2/rUSLEidQ1Eqaz//DKmxZMQNwIDAQABo1MwUTAPBgvNVHRMBAf8E
BTADAQH/MB8GA1UdIwQYMBaAFIdZ6nzS6mvONymq88pFq5bYFUslMB0GA1UdDgQW
BBTAvnpE43mxSxi38MpC231xXAIfjDANBgkqhktG9w0BAQsFAAACQAQEAkKxxYBR
tI4e0VP18C0tHodE1jA7UPkIbyZckF4dZ6f506cT+eqgp8nrqs4nH4T6CcCQHET5
BOH6nYhRT0+c15lwdBeXCmoT4VfG6UvtPPPnBCDav3IW9bS7kRoV0Xt5fq6XowE
EKQeX Eg62uzaqGIGTqHeQfKUWnK6K2yWCB6gmEhHPul8YFMvZ3GSvXAyg/oKD1cX
kBm1Wig8uETC5QV748XbHpEVvxEIb612Ly7ld70U9aHzg1+De4+YQ61Eg/qzjn
7/FB9XxmCL3h/qt/4ILRp0yQjS3Q93TD4xrIkICCPaTlF58N/BhAMsvc3XPSf06
yVjhApHYYhFLPw==
-----END CERTIFICATE-----

```

5. Go back to the vManage GUI, then in the top-right corner, click 'Install Certificate', paste the granted certificate that was copied, and then click 'Install'



6. Thus, the vSmart device has been successfully installed on vManage or successfully onboarded

The screenshot shows a log entry for the 'Install Certificate' task. It indicates a success status with the message: 'Successfully synced vEdge list on vSmart'. Below this, several log entries show the process of updating controllers with a new certificate serial number and installing the certificate for the vSmart device.

7. DO THE SAME FOR VSMART2

8. Thus, all controllers have successfully gone 'In sync' or been onboarded with vManage.

The screenshot displays the 'WAN Edge List' section under 'CONFIGURATION | DEVICES'. It lists four controllers: vManage, vSmart1, vSmart2, and vBond. All controllers are marked as 'In Sync' and have their 'Certificate Status' listed as 'Installed'.

5. Verify the onboarded controller.

```
// vManage & vSmart
vManage(vSmart)# show control connections
```

INDEX ATION	PEER TYPE	PEER PROT	PEER SYSTEM REMOTE	CONFIGURED		SITE ID	DOMAIN ID	PRIVATE IP
				SYSTEM COLOR	IP STATE UPTIME			
0	vsmart	dtls	55.55.10.3 default	55.55.10.3 up 0:00:17:32	10	1	100.10.1.13	
0	vsmart	dtls	55.55.10.4 default	55.55.10.4 up 0:00:10:17	10	1	100.10.1.14	
0	vbond	dtls	55.55.10.2 default	55.55.10.2 up 0:01:12:33	0	0	100.10.1.12	
1	vbond	dtls	0.0.0.0 default	- up 0:01:12:33	0	0	100.10.1.12	
2	vbond	dtls	0.0.0.0 default	- up 0:01:12:35	0	0	100.10.1.12	
3	vbond	dtls	0.0.0.0 default	- up 0:01:12:35	0	0	100.10.1.12	

```
// vBond
```

```
vBond# show orchestrator connections
```

INSTANCE	PEER	TYPE	PEER	PROTOCOL	SITE	DOMAIN	PEER
				SYSTEM IP	ID	ID	PRIVATE IP
		UPTIME					
0	vsmart	dtls	55.55.10.3	0:00:12:16	10	1	100.10.1.13
0	vsmart	dtls	55.55.10.3	0:00:12:16	10	1	100.10.1.13
0	vsmart	dtls	55.55.10.4	0:00:05:13	10	1	100.10.1.14
0	vsmart	dtls	55.55.10.4	0:00:05:13	10	1	100.10.1.14
0	vmanage	dtls	55.55.10.1	0:01:07:16	10	0	100.10.1.11
0	vmanage	dtls	55.55.10.1	0:01:07:16	10	0	100.10.1.11
0	vmanage	dtls	55.55.10.1	0:01:07:16	10	0	100.10.1.11
0	vmanage	dtls	55.55.10.1	0:01:07:16	10	0	100.10.1.11

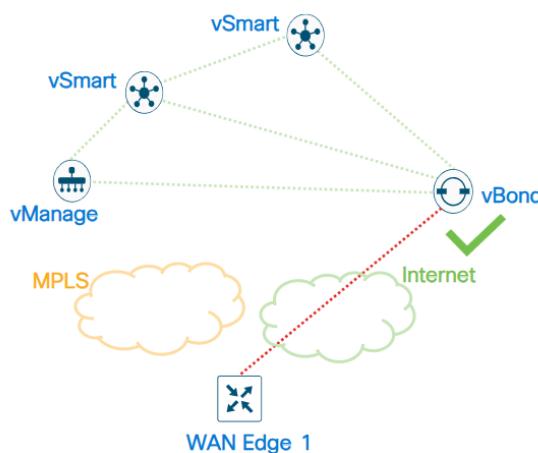
INSTALL CERTIFICATE & ONBOARDING

SD-WAN EDGE / VEDGE

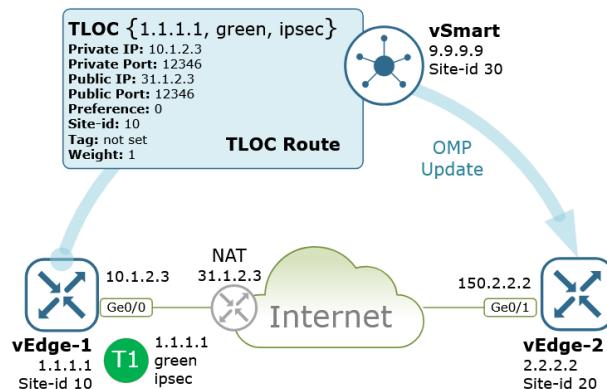
Via Public Internet

In this scenario, we will use two transport links, namely Internet and MPLS, to interconnect the vEdge to the controllers. The vEdge automatically establishes a DTLS tunnel to the vSmart and forms an OMP neighborship to exchange routing information.

During the onboarding process of the vEdge device to the controller, the vEdge device sends the 'Transport Locator (TLOC)' to the vBond (Orchestrator) when the device joins the SD-WAN fabric.



TLOC is a unique identifier used to identify the transport link that connects the vEdge device to the SD-WAN fabric.



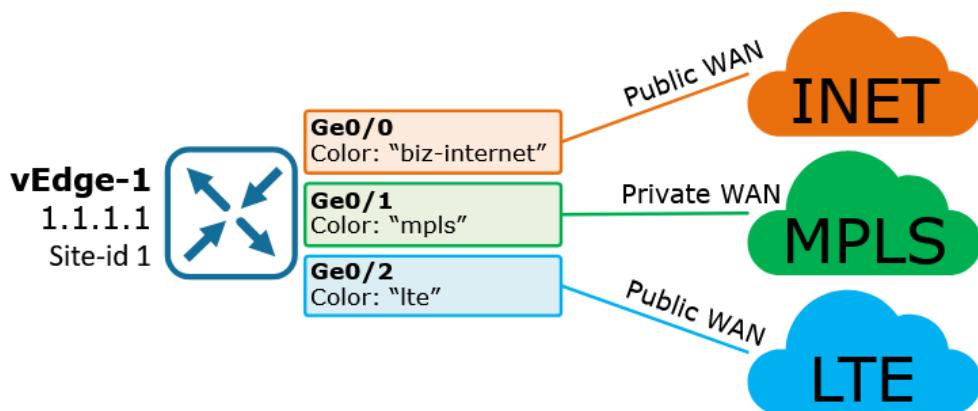
TLOC consists of several components, namely:

1. System IP

Similar to the OSPF/BGP router-ID, it functions as the identifier for the vEdge device.

2. Transport Color

The color here is used to differentiate between various transport links such as MPLS, Internet, LTE, Metro-E, etc. If the same transport link is used, colors like silver/gold can be utilized to distinguish them.



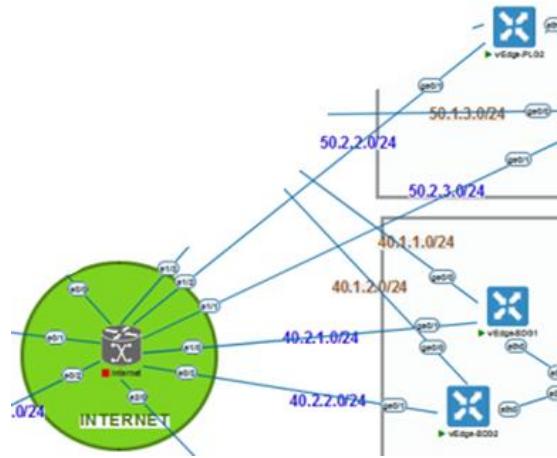
3. Encapsulation Type

This is essential for advertising data plane connectivity or communication between vEdge devices, with two options available: IPsec or GRE.

After vBond receives the TLOC from vEdge, it forwards this information to vSmart (Controller), which is responsible for managing routing and policies. Once the TLOC is initiated and registered, vSmart handles the distribution of the TLOC to other SD-WAN devices to enable inter-site connectivity.

Step by step:

*note: The configuration process is the same for every vEdge. As an example, we will onboard 'vEdge-PLG02'.



1. Setup Basic Config vEdge (system, VPN0, VPN512)

```

system
host-name vEdge-PLG02
system-ip 55.55.55.51
site-id 50
organization-name MRC
clock timezone Asia/Jakarta
vbond 100.10.1.12

vpn 0
interface ge0/0
ip address 50.1.2.1/24
tunnel-interface
encapsulation ipsec
color mpls
allow-service all
!
no shutdown

```

```
!
interface ge0/1
    ip address 50.2.2.1/24
    tunnel-interface
    encapsulation ipsec
    color public-internet
    allow-service all
!
no shutdown
!
ip route 0.0.0.0/0 50.2.2.2
!
commit
```

2. Download & Install Root Certificate from Root_CA

```
// Download root-certificate
vEdge-PLG02# request download tftp://100.10.1.100/PKI.ca
----- If it is not possible, download via vshell -----
vEdge-PLG02# vshell
vEdge-PLG02:~$ tftp -g -r PKI.ca 100.10.1.100

// Install root-certificate
vEdge-PLG02# request root-cert-chain install home/admin/PKI.ca
```

3. Generate CSR from vEdge

```
// Create CSR from vEdge device with given name 'csr.txt'
vEdge-PLG02# request csr upload home/admin/csr.txt
Uploading CSR via VPN0
Enter organization-unit name : MRC < insert organization name
Re-enter organization-unit name : MRC
```

```

Generating private/public pair and CSR for this vedge device
Generating CSR for this vedge device .....[DONE]
Copying ... /home/admin/cst.txt via VPNO
CSR upload successful

// Copy CSR that recently created
vEdge-PLG02# vshell
vEdge-PLG02:~$ ls
PKI.ca archive_id_rsa.pub csr.txt
vEdge-PLG02:~$ more csr.txt < open csr.txt file, then manually copy CSR

```

4. Request Signing to Root_CA & Paste granted certificate CSR into a new file

```

// Request the Root_CA to approve (sign) the CSR of the vEdge.
1. Go to the Root_CA device, then type 'crypto pki server PKI request pkcs10 pem terminal' and paste the CSR of vEdge-PLG02 that has been copied.
2. Type 'quit', and the granted certificate will be issued by Root_CA. Copy the granted certificate.

// Create new txt file, then paste granted certificate that already copied
vEdge-PLG02:~$ cat << "" > cert.txt << custom file
> **paste here**

// Install certificate from new file that contain granted certificate previously
vEdge-PLG02# request certificate install home/admin/cert.txt
Installing certificate via VPN 0
Successfully installed the certificate

** The certificate installed above generates a verified 'chassis & serial number' **

```

5. Add vEdge to Controller (Onboard vEdge based Chassis & Serial num)

```
// Copy chassis-number and serial-number from vEdge
vEdge-PLG02# show certificate serial
Chassis number: 55d1e2ba-caa0-4466-bc89-80971bcc383f serial number: 07

// Add vedge chassis and serial number to vManage & vBond
vManage# request vedge add chassis-num 55d1e2ba-caa0-4466-bc89-80971bcc383f serial-num 07
vBond# request vedge add chassis-num 55d1e2ba-caa0-4466-bc89-80971bcc383f serial-num 07
```

// Verification

1. Go to the vManage GUI, under Configuration > Device, perform a refresh
2. As a result, the vEdge will be successfully added to the controller

State	Device Model	Chassis Number	Serial No./Token
●	vEdge Cloud	55d1e2ba-caa0-4466-bc89-80971bcc3...	07

// Send vEdge certificate to Controller via vManage GUI

1. Go to Configuration > Certificate, on left side 'Send to Controllers'.

St...	Device Model	Chassis Number	Hostname	IP Address	Serial No./Token
●	vEdge Cloud	55d1e2ba-caa0-4466-bc89-80971bcc383f	vEdge-PLG02	55.55.55.51	07

2. Then, certificate of the vEdge will pushed to controllers

Status	Message	Device Type	Hostname	System IP	Site ID	vManage IP
> ● Success	No new updates to be sent to ...	vBond	vBond	55.55.10.2	10	55.55.10.1
> ● Success	No new updates to be sent to ...	vManage	vManage	55.55.10.1	10	55.55.10.1
> ● Success	No new updates to be sent to ...	vSmart	vSmart1	55.55.10.3	10	55.55.10.1
> ● Success	No new updates to be sent to ...	vSmart	vSmart2	55.55.10.4	10	55.55.10.1

3. On vSmart, type ‘show control connections’. If the vEdge appears in the connection list, it means the vEdge onboarding process has been successful.

```
vSmart1# show control connections

      PEER    PEER PEER      SITE      DOMAIN PEER
INDEX TYPE   PROT SYSTEM IP     ID       ID      PRIVATE IP
UPTIME

-----
0     vedge   dtls 55.55.55.51   50      1       50.2.2.1
0:00:01:07
0     vsmart  dtls 55.55.10.4   10      1       100.10.1.14
0:02:42:03
0     vbond   dtls 0.0.0.0     0       0       100.10.1.12
0:02:42:22
0     vmanage dtls 55.55.10.1   10      0       100.10.1.11
0:02:42:20
1     vbond   dtls 0.0.0.0     0       0       100.10.1.12
0:02:42:24

vSmart1# show omp peers
R -> routes received
I -> routes installed
S -> routes sent

      PEER      TYPE      DOMAIN ID      OVERLAY ID      SITE ID      STATE      UPTIME      R/I/S
55.55.10.4    vsmart    1          1          10      up        0:02:42:11    0/0/0
55.55.55.51    vedge     1          1          50      up        0:00:01:14    0/0/0
```

6. Verifikasi Onboarded vEdge

// vManage GUI > Configuration > Device

Perform the same process steps for all vEdges, and as a result, here is the outcome of all the onboarded vEdges.

WAN Edge List							
Controllers							
Change Mode Upload WAN Edge List Export Bootstrap Configuration Sync Smart Account							
<input type="text"/> Search Options							
State	Device Model	Chassis Number	Serial No./Token	Hostname	System IP	Site ID	
Green	vEdge Cloud	55d1e2ba-caa0-4466-bc89-80971bcc3...	07	vEdge-PLG2	55.55.55.51	50	
Green	vEdge Cloud	79f7d58c-0e3a-400e-bc94-3f130ae7ffce	08	vEdge-PLG3	55.55.55.52	50	
Green	vEdge Cloud	1313fed9-0a07-4207-815a-1d81d01d1...	0D	vEdge-PLG1	55.55.55.50	50	
Green	vEdge Cloud	4a797cfcc9d-4622-8d43-4d085cc078...	0F	vEdge-BD2	55.55.55.41	40	
Green	vEdge Cloud	06bb0308c-9787-4538-84dc-dca2689e3...	10	vEdge-DC-DCI	55.55.55.30	30	
Green	vEdge Cloud	c8498490-12db-43be-85b5-af2d2e12a...	11	vEdge-DRC-Cyber2	55.55.55.20	20	
Green	vEdge Cloud	efc5928-f9f6-48ae-bca4-8a7873e51a5b	14	vEdge-HO	55.55.55.1	1	
Green	vEdge Cloud	c78b032b-1d3e-44e0-b833-38b8e4b72...	0E	vEdge-BDG1	55.55.55.40	40	

// vSmart1# show omp peers

```
vSmart1# show omp peers
R -> routes received
I -> routes installed
S -> routes sent

      PEER      TYPE      DOMAIN ID      OVERLAY ID      SITE ID      STATE      UPTIME      R/I/S
55.55.10.4    vsmart    1          1          10      up        0:18:08:08    0/0/0
55.55.55.1    vedge     1          1          1       up        0:00:13:23    0/0/0
55.55.55.20   vedge     1          1          20      up        0:00:14:12    0/0/0
55.55.55.30   vedge     1          1          30      up        0:00:14:16    0/0/0
55.55.55.40   vedge     1          1          40      up        0:00:00:48    0/0/0
55.55.55.41   vedge     1          1          40      up        0:00:14:22    0/0/0
55.55.55.50   vedge     1          1          50      up        0:13:38:32    0/0/0
55.55.55.51   vedge     1          1          50      up        0:15:27:12    0/0/0
55.55.55.52   vedge     1          1          50      up        0:14:16:37    0/0/0
```

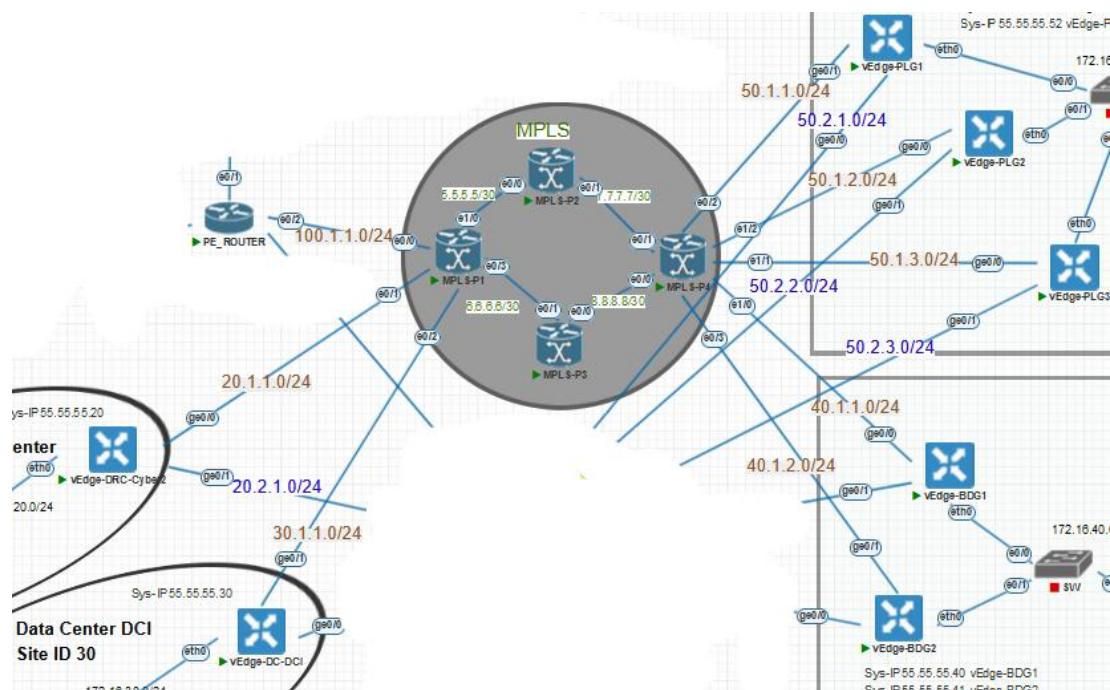
BRING-UP MPLS TRANSPORT

In the previous vEdge onboarding process to the controller, we used the public internet for the transport link. In this scenario, we will also enable the MPLS transport link as redundancy and traffic segmentation between the vEdge and the controller.

When using multiple transport links for redundancy and high availability, there is a protocol known as **Bidirectional Forwarding Detection (BFD)**. This is used to monitor the real-time condition of the underlying network, and if there is a failure in the transport link, it will detect it very quickly. BFD is enabled by default on vEdge, so if there are multiple transport links, a BFD session will be automatically established.

To enable MPLS transport to the SD-WAN fabric, **we just need to establish reachability between the vEdge and the controller**. Once the vEdge is connected to the controller, the MPLS transport link will be operational within the SD-WAN fabric.

Step by step:



1. Setup OSPF point-to-point on MPLS-P1 MPLS-P2 (link ke vEdge)

PE_Router

```
interface Ethernet0/2
ip ospf network point-to-point
```

MPLS-P1

```
interface range Ethernet0/0/0-2
ip ospf network point-to-point
```

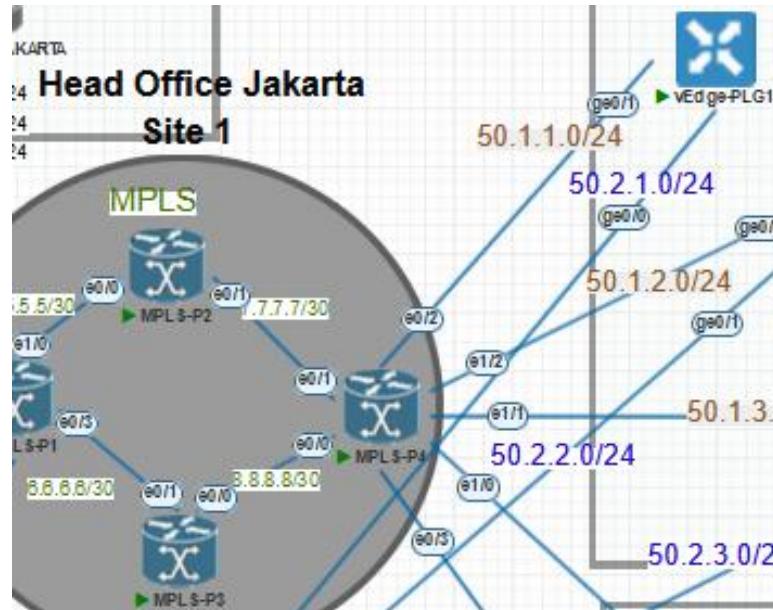
MPLS-P4

```
interface range Ethernet0/2-3, Ethernet1/0-2
ip ospf network point-to-point
```

2. Configure OSPF on all vEdge (for reachability to controllers through MPLS)

****notes: konfigurasinya sama pada semua vEdge**

// sebagai contoh disini kita akan mengkonfigurasi vEdge-PLG1



// Peering OSPF pada vEdge-PLG1

```
vpn 0
```

```
router ospf
```

```
router-id 55.55.55.50
```

```
area 0
```

```
interface ge0/1
```

```
network point-to-point
```

```
exit
```

```
exit
```

```
!
```

3. Verification OSPF reachability to controller

```
// show ospf neighbor
```

```
vEdge-PLG1# show ospf neighbor
DBsmL -> Database Summary List
RqstL -> Link State Request List
RXmtL -> Link State Retransmission List
      SOURCE
VPN   IP ADDRESS      INTERFACE      ROUTER ID      STATE
---  -----
0     50.1.1.2          ge0/1          2.2.2.4        full
```

```
// show ip route
```

VPN	PREFIX	PROTOCOL	PROTOCOL SUB TYPE	NEXTHOP IF NAME	NEXTHOP ADDR
0	0.0.0.0/0	static	-	ge0/0	50.2.1.2
0	1.1.1.0/24	ospf	IA	ge0/1	50.1.1.2
0	2.2.2.1/32	ospf	IA	ge0/1	50.1.1.2
0	2.2.2.2/32	ospf	IA	ge0/1	50.1.1.2
0	2.2.2.3/32	ospf	IA	ge0/1	50.1.1.2
0	2.2.2.4/32	ospf	IA	ge0/1	50.1.1.2
0	5.5.5.0/30	ospf	IA	ge0/1	50.1.1.2
0	6.6.6.0/30	ospf	IA	ge0/1	50.1.1.2
0	8.8.8.0/30	ospf	IA	ge0/1	50.1.1.2
0	30.1.1.0/24	ospf	IA	ge0/1	50.1.1.2
0	40.1.1.0/24	ospf	IA	ge0/1	50.1.1.2
0	40.1.2.0/24	ospf	IA	ge0/1	50.1.1.2
0	50.1.1.0/24	ospf	IA	ge0/1	-
0	50.1.1.0/24	connected	-	ge0/1	-
0	50.1.2.0/24	ospf	IA	ge0/1	50.1.1.2
0	50.2.1.0/24	connected	-	ge0/0	-
0	55.55.55.50/32	connected	-	system	-
0	100.1.1.0/24	ospf	IA	ge0/1	50.1.1.2
0	100.2.1.0/24	ospf	IA	ge0/1	50.1.1.2
0	100.10.1.0/24	ospf	IA	ge0/1	50.1.1.2

```
// show control connections (make sure MPLS STATE UP)
```

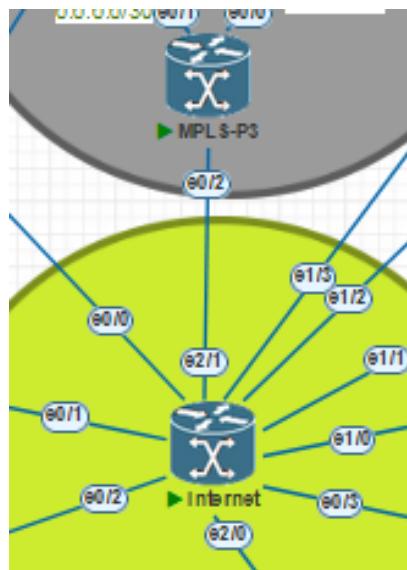
TYPE	PEER PROT	PEER SYSTEM IP	SITE ID	DOMAIN ID	PEER PRIVATE IP	PEER PORT	PEER PRIV PORT	PEER PUBLIC IP	PEER LOCAL COLOR	PEER PROXY STATE
vsmart	dts	55.55.10.3	10	1	100.10.1.13	12446	100.10.1.13		12446	mpls No up
vsmart	dts	55.55.10.4	10	1	100.10.1.14	12446	100.10.1.14		12446	mpls No up
vsmart	dts	55.55.10.3	10	1	100.10.1.13	12446	100.10.1.13		12446	public-internet No up
vsmart	dts	55.55.10.4	10	1	100.10.1.14	12446	100.10.1.14		12446	public-internet No up
vmanage	dts	55.55.10.1	10	0	100.10.1.11	12446	100.10.1.11		12446	public-internet No up

```
// show bfd session
```

It seems that the connection between the MPLS transport link and the public internet is still being detected as down. **This is because the two links do not have reachability** (for example, the segment leading to MPLS from vEdge-PLG1 cannot ping the segment leading to the public internet from vEdge-BDG1).

SYSTEM IP TRANSITIONS	SITE ID	STATE	SOURCE TLOC		REMOTE TLOC	
			COLOR	COLOR	SOURCE IP	COLOR
55.55.55.1	1	up	mpls	metro-ethernet	50.1.1.1	2
55.55.55.1	1	up	public-internet	metro-ethernet	50.2.1.1	2
55.55.55.20	20	down	mpls	public-internet	50.1.1.1	0
55.55.55.20	20	up	public-internet	public-internet	50.2.1.1	0
55.55.55.30	30	down	mpls	public-internet	50.1.1.1	0
55.55.55.30	30	up	public-internet	public-internet	50.2.1.1	0
55.55.55.40	40	down	mpls	public-internet	50.1.1.1	0
55.55.55.40	40	up	public-internet	public-internet	50.2.1.1	0
55.55.55.41	40	down	mpls	public-internet	50.1.1.1	0
55.55.55.41	40	up	public-internet	public-internet	50.2.1.1	0

4. Connect MPLS & Internet to Bring UP Full BFD Session/Reachability



We will use OSPF to advertise routes connected to the internet, so that both transport links have reachability and can activate the BFD session.

**** notes:** For example, vEdge-PLG1 communicates with vEdge-BDG1 using the same transport link, MPLS (MPLS to MPLS). However, if one of the MPLS links fails, BFD will act quickly and switch to the public internet transport link (MPLS to Public Internet).

// INTERNET

```
interface Ethernet2/1
ip address 3.3.3.1 255.255.255.252
ip ospf network point-to-point
shutdown
duplex auto
!
router ospf 1
router-id 3.3.3.3
network 0.0.0.0 255.255.255.255 area 0
```

!

// MPLS-P3

```
interface Ethernet0/2
```

```
ip address 3.3.3.2 255.255.255.252
```

```
ip ospf network point-to-point
```

```
duplex auto
```

```
end
```

```
!
```

```
** OSPF sudah aktif di MPLS-P3
```

// Semua BFD Session antara MPLS dan Public Internet menjadi aktif

vEdge-PLG1# show bfd sessions		SYSTEM IP	SITE ID	STATE	SOURCE TLOC COLOR	REMOTE TLOC COLOR	SOURCE IP
NS							
55.55.55.1	1	up	mpls		metro-ethernet		50.1.1.1
55.55.55.1	1	up	public-internet	metro-ethernet			50.2.1.1
55.55.55.20	20	up	mpls		public-internet		50.1.1.1
55.55.55.20	20	up	public-internet	public-internet			50.2.1.1
55.55.55.30	30	up	mpls		public-internet		50.1.1.1
55.55.55.30	30	up	public-internet	public-internet			50.2.1.1
55.55.55.40	40	up	mpls		public-internet		50.1.1.1
55.55.55.40	40	up	public-internet	public-internet			50.2.1.1
55.55.55.41	40	up	mpls		public-internet		50.1.1.1
55.55.55.41	40	up	public-internet	public-internet			50.2.1.1

VMANAGE TEMPLATE OVERVIEW

Device Template & Feature Template

Previously, we used CLI for all device configurations. CLI is a good way to quickly configure something, but it is not scalable. CLI is suitable for configuring a few devices, but if we need to handle dozens of devices, it will take too much time and could lead to errors. An alternative way to address this weakness is by using templates.

One of the powerful features in SD-WAN is the ability to provide templates. With this feature, we simply create a set of configuration templates and push them to the devices. We create the template first and then apply it to one or as many devices as needed. This feature is scalable, saving time and reducing the likelihood of configuration errors.

There are 2 types of templates, namely:

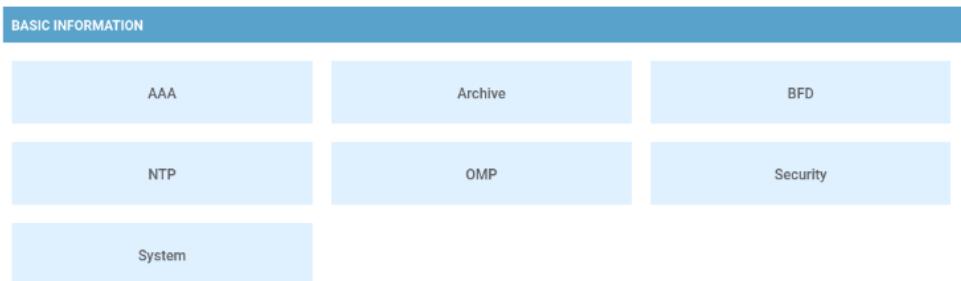
Device template

This can be considered a collection of several feature templates, where a specific device model can be associated with a device template (such as Cisco ISR, CSR, Catalyst 8000 series, vEdge 100, vEdge 200, etc.). This is the template that will be attached to the device.

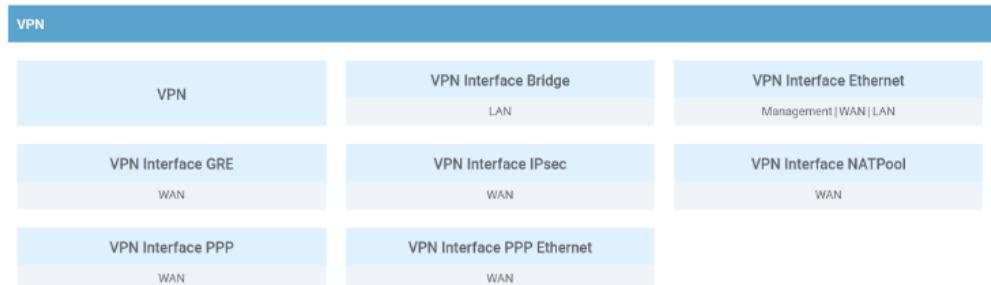
Feature template

We create a feature template for each 'feature' that we want to configure. One feature template can be used across multiple device templates. Feature templates have three categories, namely:

a. Basic Configuration



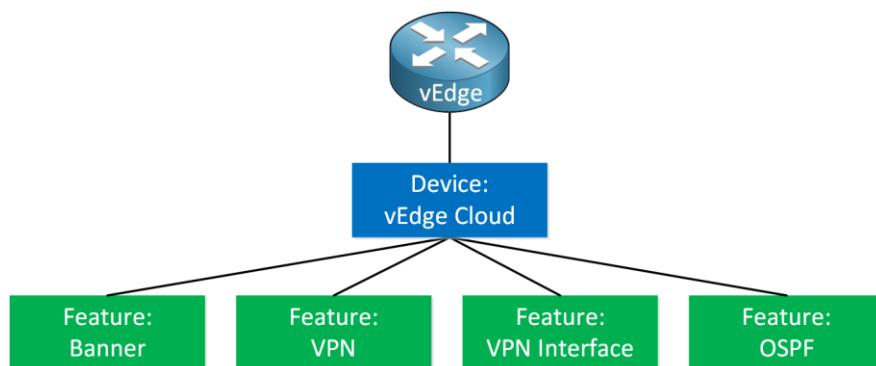
b. VPN



c. Other Templates

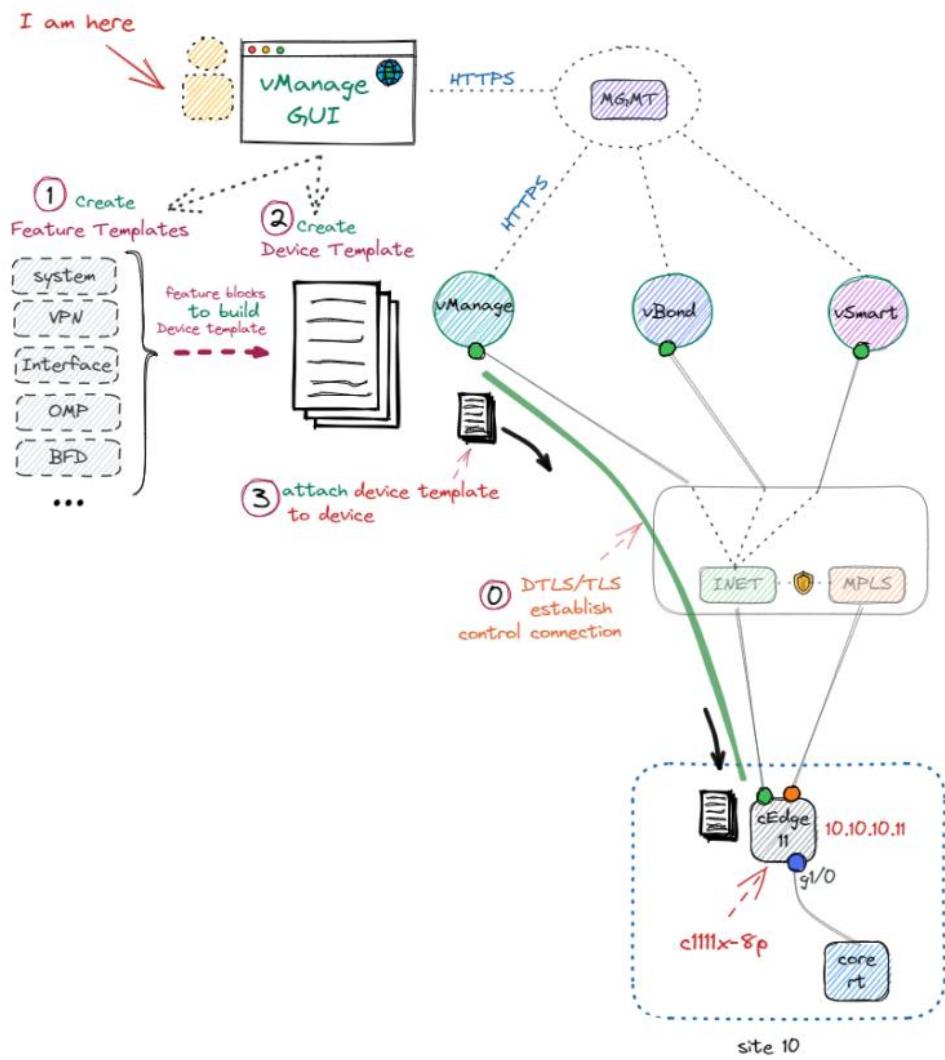


There is a 'parent-child' relationship between the Device template and the Feature template. First, we create the Feature template and place it into the Device template. Then, we attach the Device template to the device, such as the vEdge Cloud router.



How the templates are created and deployed

CISCO SDWAN Feature Template



Source: <https://blogs.itbase.tv/part-10-cisco-sd-wan-configuration-templates>

The user configures the template through vManage, creating a collection of feature templates that will be assigned to the device template. Once the device template and the collection of feature templates are ready, the device template is pushed/attached to the vEdge via the DTLS tunnel. As a result, the vEdge is now in vManage mode. It is important to note that **once a vEdge is set to vManage mode (via template), it can no longer be configured using CLI mode.**

Device Template Method

In template configuration, a device template can be attached with a set of predefined feature templates. **A device template can be attached to multiple vEdges simultaneously, but if one vEdge wants to add a new parameter, other vEdges attached to that device template will also be affected.**

Templates are created to overcome the limitations of CLI mode, offering flexibility and ease in applying configurations. There are several methods that can be used to flexibly create templates for vEdges according to specific needs, such as:

1. Initial templates

These are the initial templates that can be applied to all vEdges when joining the SD-WAN fabric. They consist of common configurations shared by all vEdges in their interconnection to the SD-WAN fabric, such as VPN0, System, OSPF/BGP for MPLS transport link, default route to the public internet, etc.

2. Specific device template

This is for specific devices that require special configurations, such as having multiple service VPNs, specific policies, or custom features.

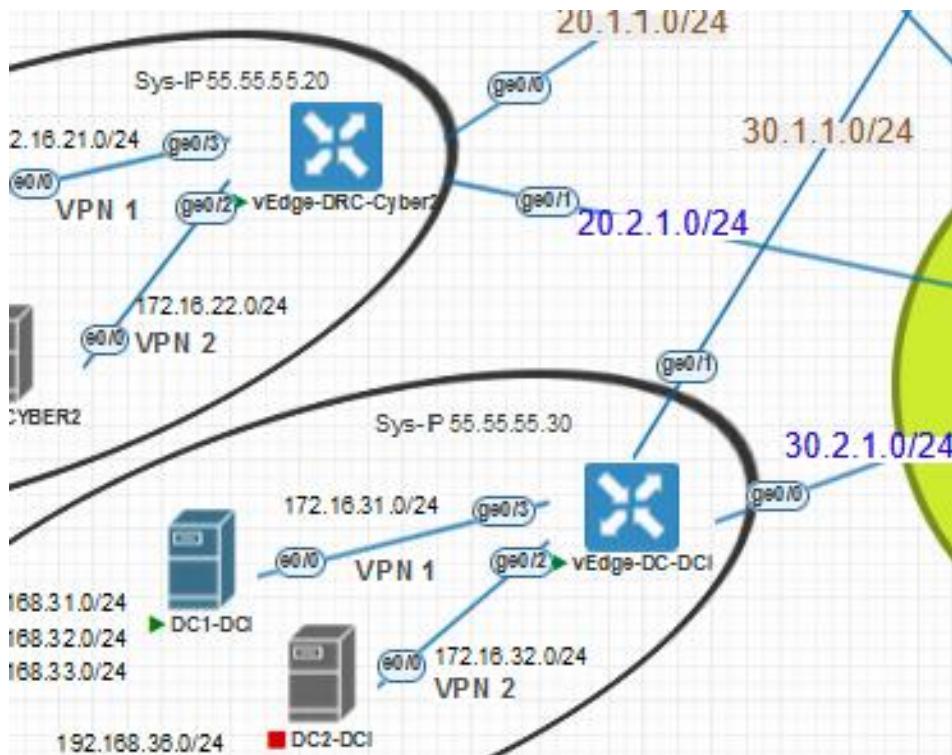
3. One transport link template (MPLS only / Internet only)

This can be used for vEdges that have only a single transport link, either MPLS or Internet.

4. Custom templates

Custom templates allow users to define specific configurations tailored to particular needs or use cases within the SD-WAN network

Step by step:



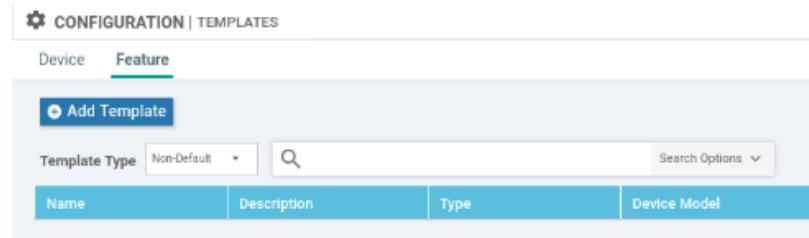
We will attempt to configure vEdge-DC-DCI and vEdge-DRC-Cyber2 through vManage Device Template. They have VPN 0 along with OSPF routing for MPLS and a default route for the internet, VPN 1, and VPN 2, for which we will create feature templates. However, the focus in this section will be on configuring VPN 0 only, while the Service VPN will be covered in another section. Additionally, we will also configure System and Banner, and then in the same device template, both devices will be attached to the template.

1. Create and setup Features template

In the feature template we will create, our focus will be on VPN 0 for both devices. There are 4 sections for the 'vEdge Cloud' device we will use: 'Basic Configuration', 'VPN', and 'Other Template'. In the configuration scenario, we will create the following feature templates:

- **Basic Configuration:** System
- **VPN:** VPN (VPN 0 & VPN 512), VPN Interface (Ge0/0, Ge0/1, Eth0)
- **Other Template:** Banner, OSPF (MPLS interconnection)

1. Go to vManage GUI, Configuration > Templates > Features, then select 'Add Template'



2. On 'Select Device' section, choose the desired device model, here we are using 'vEdge Cloud', then on the right you will see the templates that can be used.

3. We will start with the 'System' template, click on the template and give it a name and description.

***quick notes:**

There are 3 options that can be used:

- Global (Globe icon)

This is applied to all devices using this feature template. For example, VPN ID or Console Baud Rate.

- Default (Checklist icon)

This has several options predefined by vManage, and can be selected, such as timezone or Overlay ID.

- Device specific (Radio icon)

We can apply this to several devices that have their own values, such as Site ID and System IP. This option is set in the form of variables, which will later be filled by the specific device (e.g., [system_site_id])

4. In the ‘Basic Configuration’ section, fill in according to the intended template usage, then ‘Save’. This system template is now ready to be used for the Device template

The screenshot shows the 'CONFIGURATION | TEMPLATES' page with the 'Feature' tab selected. A blue button labeled 'Add Template' is visible. Below it, there's a search bar with 'Template Type' set to 'Non-Default'. A table lists one template entry:

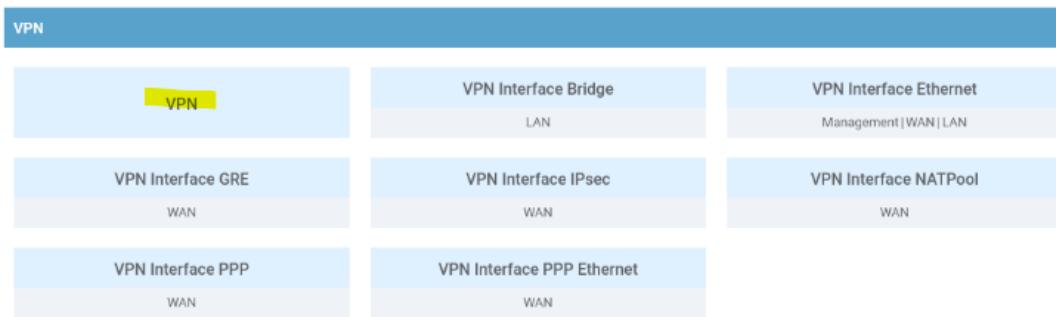
Name	Description	Type	Device Model
vEdge_System_Template	vEdge_System_Template	WAN Edge System	vEdge Cloud

5. Other template > Banner Templates, save

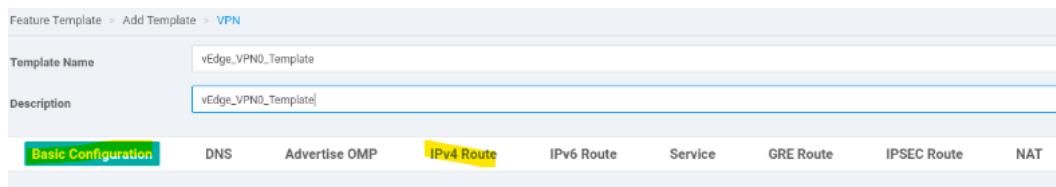
The screenshot shows the 'Feature Template > Add Template > Banner' configuration page. It includes fields for 'Device Type' (vEdge Cloud), 'Template Name' (vEdge_Banner_Template), and a 'Description' field which is marked as required. The 'BASIC CONFIGURATION' section contains two banner configurations:

- Login Banner:** A dropdown menu is open, showing 'AUTOHORIZED USER ONLY'.
- MOTD Banner:** A dropdown menu is open, showing 'WELCOME TO MRC.ORG'.

6. VPN > VPN



7. On 'VPN' template, there is several section as follows



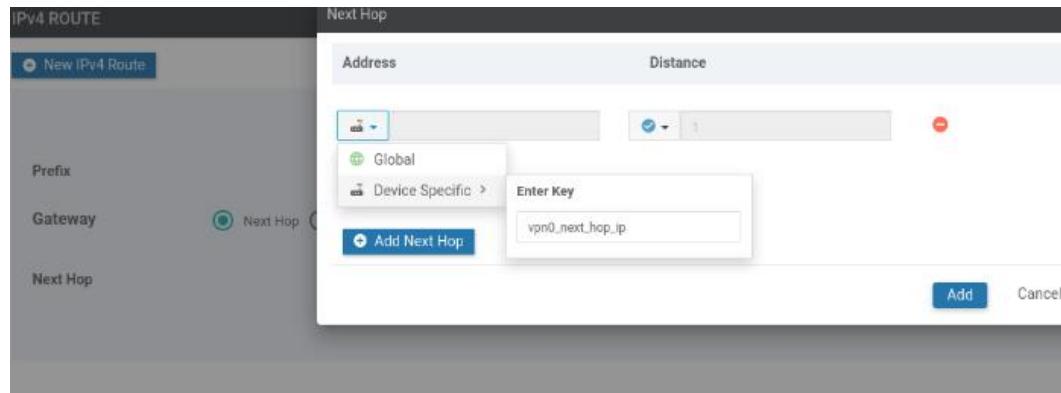
8. [VPN 0 Templates]

Here we are only configuring ‘Basic Configuration’ and ‘IPv4 Route’ as the transport link, but for VPN1 and VPN2, the ‘Advertise OMP’ section needs to be configured based on the routing protocol to be advertised.

9. In ‘Basic Configuration’, select the **Global** option for VPN and Name, then set it to 0 (VPN0) and name the VPN as ‘Transport VPN’

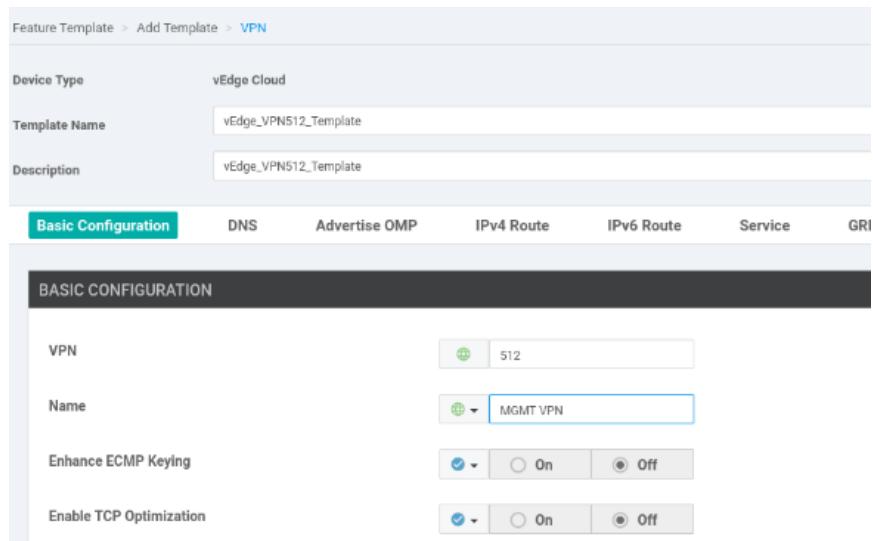
VPN	0
Name	Transport VPN
Enhance ECMP Keying	Off
Enable TCP Optimization	Off

10. Scroll down to the ‘IPv4 routes’ section, click ‘New IPv4 Route’. We will use the **Global** option for Prefix since we will set the default route to the Internet (0.0.0.0/0). For the Gateway, choose **Next hop** Device specific option, then create the variable like the following, then ‘Add’. Finally, ‘Save’ the VPN template.

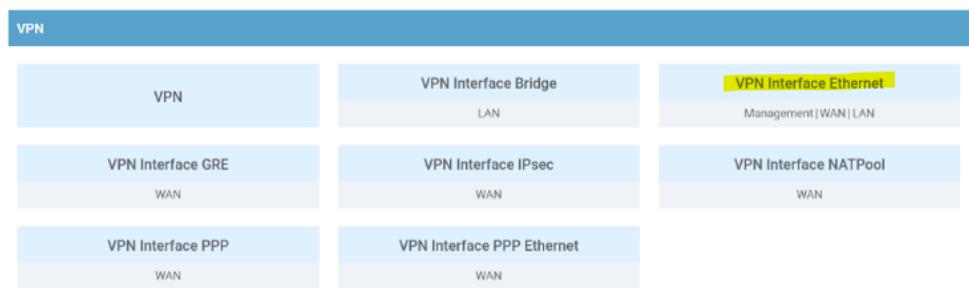


11. [VPN 512 Template]

For VPN 512, we will only configure ‘Basic Configuration’, there is no IPv4 route for this VPN



12. VPN > VPN Interface



13. [VPN 0, Interface Ge0/0]

Give the template a name, set Shutdown to ‘No’, then specify the interface name and description

Feature Template > Add Template > **VPN Interface Ethernet**

Template Name: vEdge_VPN0_Int_Ge0/0_Template

Description: vEdge_VPN0_Int_Ge0/0_Template

Basic Configuration Tunnel NAT VRRP ACL/QoS ARP 802.1X Advanced

BASIC CONFIGURATION

Shutdown: Yes No

Interface Name: ge0/0

Description: [vpn0_int_Ge0-0_description]

IPv4 **IPv6**

14. In the ‘Basic Configuration’ section, select ‘IPv4’, then Static, and change the variable for Device option in ‘IP address’

Basic Configuration Tunnel NAT VRRP ACL/QoS ARP 802.1X Advanced

IPv4 **IPv6**

Dynamic Static

IPv4 Address: [vpn0_int_Ge0-0_ipv4_address]

Secondary IP Address (Maximum: 4): Add

DHCP Helper:

Block Non Source IP: Yes No

Bandwidth Upstream:

Bandwidth Downstream:

15. Since this is for the transport link, enable ‘Tunnel interface’, and for ‘Color’, we use Device specific with the following variable

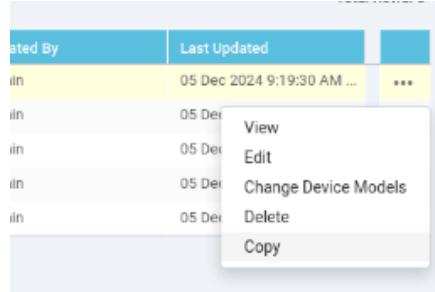
The screenshot shows a configuration panel titled "TUNNEL". It contains five sections: "Tunnel Interface" (radio button selected for "On"), "Color" (dropdown menu set to [vpn0_int_Ge0-0_tunnel_color_value]), "Restrict" (radio button selected for "Off"), "Groups" (dropdown menu), and "Border" (radio button selected for "Off").

16. Allow service all, then save

The screenshot shows a configuration panel titled "Allow Service". It has two sections: "All" (radio button selected for "On") and "BGP" (radio button selected for "Off").

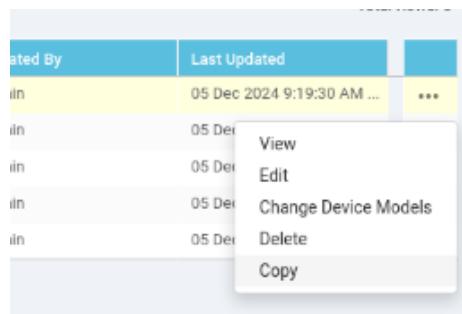
17. [VPN 0, Interface Ge0/1]

Do the same as for interface Ge0/0, or copy the Ge0/0 Template, then change according to the Ge0/1 parameters.



18. [VPN 512, Interface eth0]

Do the same as for interfaces Ge0/0 and Ge0/1, or copy the previous template, then change according to the eth0 parameters (default IP address). **For VPN 512, the tunnel interface does not need to be enabled.**



19. Other Template > OSPF (for MPLS link)



20. [OSPF Template]

Set the template name, then for ‘Router-id’, select the Device specific option

Device Type	vEdge Cloud			
Template Name	vEdge_VPN0 OSPF_Template			
Description	vEdge_VPN0 OSPF_Template			
Basic Configuration	Redistribute	Maximum Metric (Router LSA)	Area	Advanced
BASIC CONFIGURATION				
Router ID	<input type="text"/> [ospf_router_id]			
Distance for External Routes	<input checked="" type="checkbox"/> 110			
Distance for Inter-Area Routes	<input checked="" type="checkbox"/> 110			
Distance for Intra-Area Routes	<input checked="" type="checkbox"/> 110			

21. In the ‘Area’ section, define the area number, then ‘Add interface’.

Basic Configuration Redistribute Maximum Metric (Router LSA) **Area** Advanced

AREA

New Area Mark as Optional Row i

Area Number	<input type="text" value="0"/>
Set the area type	<input type="text" value=""/>
Interface	+ Add Interface
Range	+ Add Range

Add **Cancel**

22. For ‘Interface Name’, select Device specific with the variable for the interface name.

The screenshot shows the 'Add Interface' configuration page. The 'Interface Name' field is populated with the variable '[ospf_area_0_if_name]'. Other configuration options shown include 'Hello Interval (seconds)' set to 10, 'Dead Interval (seconds)' set to 40, 'LSA Retransmission Interval (seconds)' set to 5, and 'Interface Cost'.

23. Click ‘Advanced Options’, then select Point-to-Point for OSPF network type, then Save.

The screenshot shows the 'Advanced Options' configuration page. The 'OSPF Network Type' dropdown is set to 'point-to-point'.

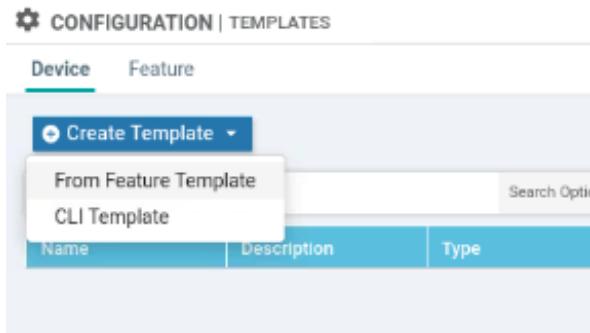
24. This way, the Feature template is ready to be used and attached to the Device Template

The screenshot shows the 'Feature' template management page. The table lists various feature templates:

Name	Description	Type	Device Model	Device Temp
vEdge_VPN512_Int_eth0_...	vEdge_VPN512_Int_eth0_T...	WAN Edge Interface	vEdge Cloud	0
vEdge_VPN0_Int_Ge0/1_T...	vEdge_VPN0_Int_Ge0/1_Te...	WAN Edge Interface	vEdge Cloud	0
vEdge_VPN0_Int_Ge0/0_T...	vEdge_VPN0_Int_Ge0/0_Te...	WAN Edge Interface	vEdge Cloud	0
vEdge_System_Template	vEdge_System_Template	WAN Edge System	vEdge Cloud	0
vEdge_VPN0_Template	vEdge_VPN0_Template	WAN Edge VPN	vEdge Cloud	0
vEdge_VPN512_Template	vEdge_VPN512_Template	WAN Edge VPN	vEdge Cloud	0
vEdge_VPN0_OSPF_Template	vEdge_VPN0_OSPF_Template	OSPF	vEdge Cloud	0
vEdge_Banner_Template	vEdge_Banner_Template	Banner	vEdge Cloud	0

2. Create and setup Device template

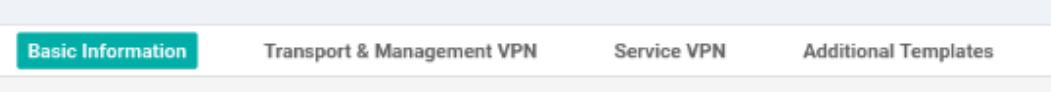
1. Go to Configuration > Templates > Device template, then click ‘Create Template’ and select ‘From Feature Template’.



2. Specify the Device model, Template Name, and description of the Device template to be created.

The screenshot shows the 'Device' tab selected in the 'Configuration | Templates' interface. It includes fields for 'Device Model' (set to 'vEdge Cloud'), 'Template Name' ('vEdge_DC-DRC_Site_Template'), and 'Description' ('vEdge_DC-DRC_Site_Template').

3. There are 4 sections in the Device template, which are:



4. [Basic Information]

Specify the Feature templates that have been created according to the template type to be associated with the Device Template.

The screenshot shows the 'Basic Information' section. Under 'System *', a dropdown menu is open, showing 'Factory_Default_vEdge_System_Template' and 'vEdge_System_Template'. The 'vEdge_System_Template' option is highlighted with a blue box. Other options include 'Create Template' and 'View Template'.

5. [Transport & Management VPN]

Add the Additional VPN 0 Template for OSPF and VPN interface.

Transport & Management VPN	
VPN 0 *	vEdge_VPN0_Template
VPN Interface	vEdge_VPN0_Init_Ge0/0_Template
Additional VPN 0 Templates	
<input checked="" type="radio"/> BGP	
<input checked="" type="radio"/> OSPF	
<input checked="" type="radio"/> VPN Interface	
<input type="radio"/> VPN Interface GRE	
<input type="radio"/> VPN Interface IPsec	
<input type="radio"/> VPN Interface PPP	
VPN 512 *	Factory_Default_vEdge_VPN_512_Template
Additional VPN 512 Templates	
<input type="radio"/> VPN Interface	

6. Associate the feature templates that have been created.

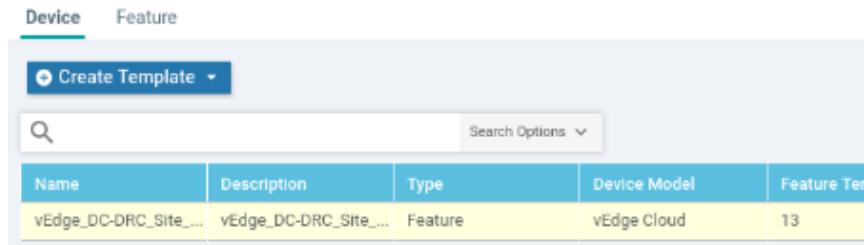
Transport & Management VPN	
VPN 0 *	vEdge_VPN0_Template
OSPF	vEdge_VPN0_OSPF_Template
VPN Interface	vEdge_VPN0_Init_Ge0/0_Template
VPN Interface	vEdge_VPN0_Init_Ge0/1_Template
VPN 512 *	vEdge_VPN512_Template
VPN Interface	vEdge_VPN512_Init_eth0_Template

7. [Additional Template]

Add the Banner from the feature template that was created, then click 'Create'.

Additional Templates	
Banner	Choose...
Policy	Choose...
SNMP	Choose...
Security Policy	Choose...
Bridge	
<input checked="" type="radio"/> Bridge	
<input type="button" value="Create"/> <input type="button" value="Cancel"/>	

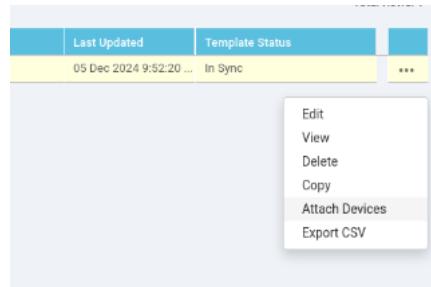
8. With that, the Device template is successfully created and ready to be attached to the device



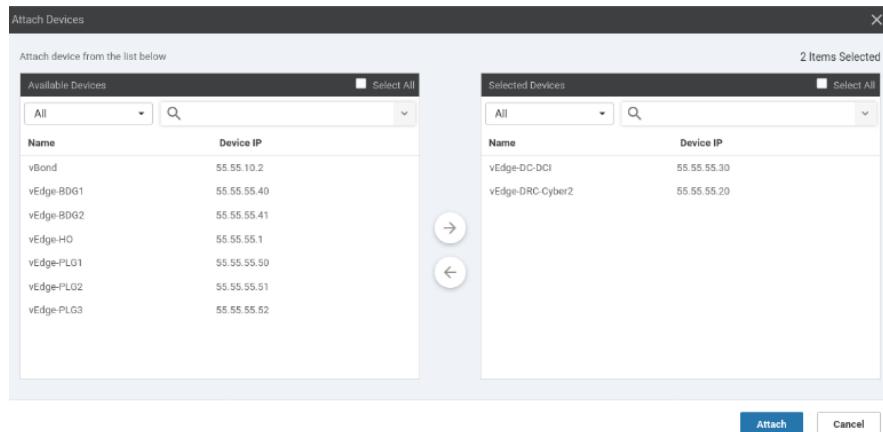
Name	Description	Type	Device Model	Feature Tier
vEdge_DC-DRC_Site_...	vEdge_DC-DRC_Site_...	Feature	vEdge Cloud	13

3. Attach Device template to Device

1. In Configuration > Template > Device Template, select the Device template that has been created, then on the far right, select ‘Attach Devices’.



2. Move DRC-Cyber and DC-DCI to the right to apply the device template that has been created, then click ‘Attach’.



3. ‘Edit Device Template’ for each device, according to the values for each device.



S.	Chassis Number	System IP	Hostname	Description(vpn0_int_Ce0_0.description)	Description(vpn0_int_Ge0_0.description)	IPv4 Address(vpn0_...	Total Rows: 2
0	06b0308c-9787-4538-84dc-dca2689e3aee	55.55.55.30	vEdge-DC-DCI				...
1	c8498490-12db-43be-85b5-af2d2e12a71b	55.55.55.20	vEdge-DRC-Cyber2				...

4. [vEdge-DC-DCI]

Update Device Template X

Variable List (Hover over each field for more information)

Chassis Number	06b0308c-9787-4538-84dc-dca2689e3aee
System IP	55.55.55.30
Hostname	vEdge-DC-DCI
Address(vpn0_next_hop_ip)	30.2.1.2
Description(vpn0_int_Ge0-1_description)	To MPLS link
IPv4 Address(vpn0_int_Ge0-1_ipv4_address)	30.1.1.2/24
Color(vpn0_int_Ge0-1_tunnel_color_value)	mpls
Description(vpn0_int_Ge0-0_description)	To Public Internet
IPv4 Address(vpn0_int_Ge0-0_ipv4_address)	30.2.1.1/24
Color(vpn0_int_Ge0-0_tunnel_color_value)	public-internet
Router ID(ospf_router_id)	30.1.1.1
Interface Name(ospf_area_0_if_name)	ge0/1
Hostname	vEdge-DC-DCI
System IP	55.55.55.30
Site ID	30

Generate Password Update Cancel

5. [vEdge-DC-DCI]

Update Device Template X

Variable List (Hover over each field for more information)

Chassis Number	c8498490-12db-43be-85b5-af2d2e12a71b
System IP	55.55.55.20
Hostname	vEdge-DRC-Cyber2
Address(vpn0_next_hop_ip)	20.2.1.2
Description(vpn0_int_Ge0-1_description)	TO PUBLIC INTERNET
IPv4 Address(vpn0_int_Ge0-1_ipv4_address)	20.2.1.1/24
Color(vpn0_int_Ge0-1_tunnel_color_value)	public-internet
Description(vpn0_int_Ge0-0_description)	To MPLS
IPv4 Address(vpn0_int_Ge0-0_ipv4_address)	20.1.1.1/24
Color(vpn0_int_Ge0-0_tunnel_color_value)	mpls
Router ID(ospf_router_id)	55.55.55.20
Interface Name(ospf_area_0_if_name)	ge0/0
Hostname	vEdge-DRC-Cyber2
System IP	55.55.55.20
Site ID	20

Generate Password Update Cancel

6. Click ‘Update’, then ‘Next’, and we can review the changes before applying them.

Device Template		Total	Config Preview		Config Diff	'Configure' action will be applied to 1 device(s) attached to 1 device template(s).	
vEdge_DC_DRC_Site_Template		1					
Device list (Total: 1 devices)							
Filter/Search							
0db0308c-9787-4538-84dc-dca2689e3aeef vEdge DC-DCI 55.55.55.30							
Local Configuration vs. New Configuration							
1	1		viptela-system:system				
2	2		device-model	vedge-cloud			
3	3		chassis-number	0db0308c-9787-4538-84dc-dca2689e3aeef			
4	3		host-name	vEdge-DC-DCI			
5	4		system-ip	55.55.55.30			
5	5		domain-id	1			
6	6		site-id	30			
7	7		admin-tech-on-failure				
8	8		no route-consistency-check				
9	9		sp-organization-name	MRC			
10	10		organization-name	MRC			
11			clock timezone	Asia/Jakarta			
12	11		vbond	100.10.1.12 port 12346			
13	12		aaa				
14	13		auth-order	local radius tacacs			
15	14		usergroup	basic			
16	15		task system	read write			
17	16		task interface	read write			

7. Once confirmed, select ‘Configure Devices’.

8. The Device template is successfully pushed to the selected devices.

Push Feature Template Configuration ✓ Validation Success				
Total Task: 1 Success : 1				
<input type="text"/> Search Options ▾				
	Status	Message	Chassis Number	Device Model
▼	✓ Success	Done - Push Feature Temp...	06b0308c-9787-4538-84d...	vEdge Cloud
		[5-Dec-2024 17:50:25 WIB] Checking and creating device in vManage		
		[5-Dec-2024 17:50:25 WIB] Device is online		
		[5-Dec-2024 17:50:25 WIB] Updating device configuration in vManage		
		[5-Dec-2024 17:50:26 WIB] Pushing configuration to device.		
		[5-Dec-2024 17:50:27 WIB] Pre-checks on vManage have passed. Continuing with pushing config		
		[5-Dec-2024 17:50:28 WIB] Pushing configuration to device. Please wait ...		

4. Verification Pushed Device Template

// Configuration > Device, the mode of both devices becomes 'vManage', meaning they are configured through vManage, with the Device template shown.

Hostname	System IP	Site ID	Mode	Assigned Template	Device Status
vEdge-PLG2	55.55.55.51	50	CLI	--	In Sync
vEdge-PLG3	55.55.55.52	50	CLI	--	In Sync
vEdge-PLG1	55.55.55.50	50	CLI	--	In Sync
vEdge-BDG2	55.55.55.41	40	CLI	--	In Sync
vEdge-DC-DCI	55.55.55.30	30	vManage	vEdge_DC-DRC_Site_Te...	In Sync
vEdge-DRC-Cyber2	55.55.55.20	20	vManage	vEdge_DC-DRC_Site_Te...	In Sync
vEdge-HO	55.55.55.1	1	CLI	--	In Sync
vEdge-BDG1	55.55.55.40	40	CLI	--	In Sync

// vEdge-DC-DCI

```

banner
  login "AUTORIZED USER ONLY"
  motd  "WELCOME TO MRC ORG"
!
vpn 0
  name "Transport VPN"
  router
    ospf
      router-id 55.55.55.30
      timers spf 200 1000 10000
      area 0
        interface ge0/1
        network point-to-point
        exit
      exit
!
!
```

// vEdge-DRC-Cyber2 (Configured vManaged banner and OSPF)

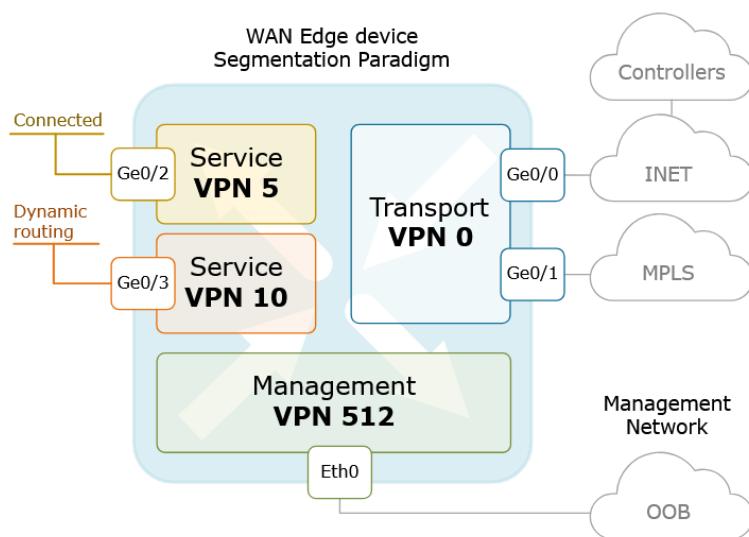
```

banner
  login "AUTORIZED USER ONLY"
  motd  "WELCOME TO MRC ORG"
!
vpn 0
  name "Transport VPN"
  router
    ospf
      router-id 55.55.55.20
      timers spf 200 1000 10000
      area 0
        interface ge0/0
        network point-to-point
        exit
      exit
!
!
```

SERVICE VPN OVERVIEW: CONNECTED & STATIC ROUTES

Via CLI and Templates

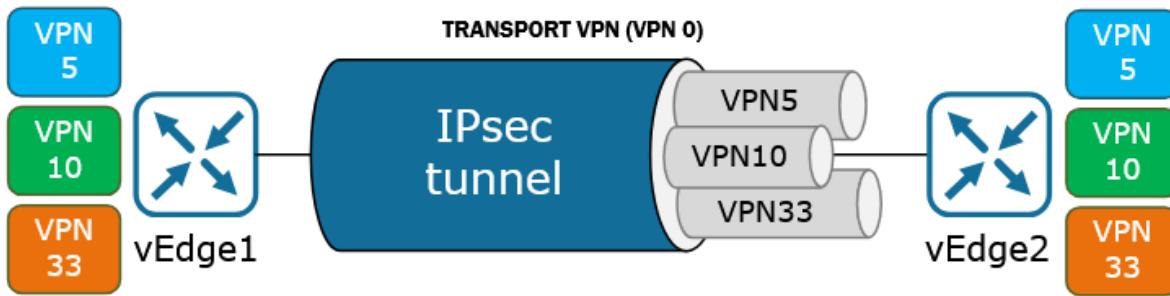
Service VPN in SD-WAN is similar to Virtual Routing Forwarding (VRF) in traditional networks. Each VPN/VRF has its own routing table, just like the segmentation done by VLAN (Layer 2), but here VPNs perform segmentation on the routing table (Layer 3).



Source: <https://www.networkacademy.io/ccie-enterprise/sdwan/vpn-segmentation>

Operationally, all controllers and vEdges are connected to each other, both from the controller to vEdge (DTLS tunnel) and from vEdge to vEdge (IPsec tunnel). All of these interconnections run on **VPN 0 / Transport VPN**, and all matters related to control plane/routing and security are handled via VPN 0.

All interfaces facing the transport link (such as MPLS/Internet) are part of the Transport VPN. They belong to their own local routing table, which we can refer to as the 'WAN Facing' routing table, while the interfaces behind the vEdge are referred to as 'LAN Facing'. All connectivity between these 'WAN Facing' interfaces occurs on VPN 0. **What makes SD-WAN shine is its ability to take all the Service VPN prefixes behind a vEdge and immediately forward them to the Service VPNs on another vEdge.**



Service VPN does not know how the packets are sent; all packet forwarding is handled by the Transport VPN. **The interconnection of Service VPNs behind the vEdge router is done just like in traditional networks, such as:**

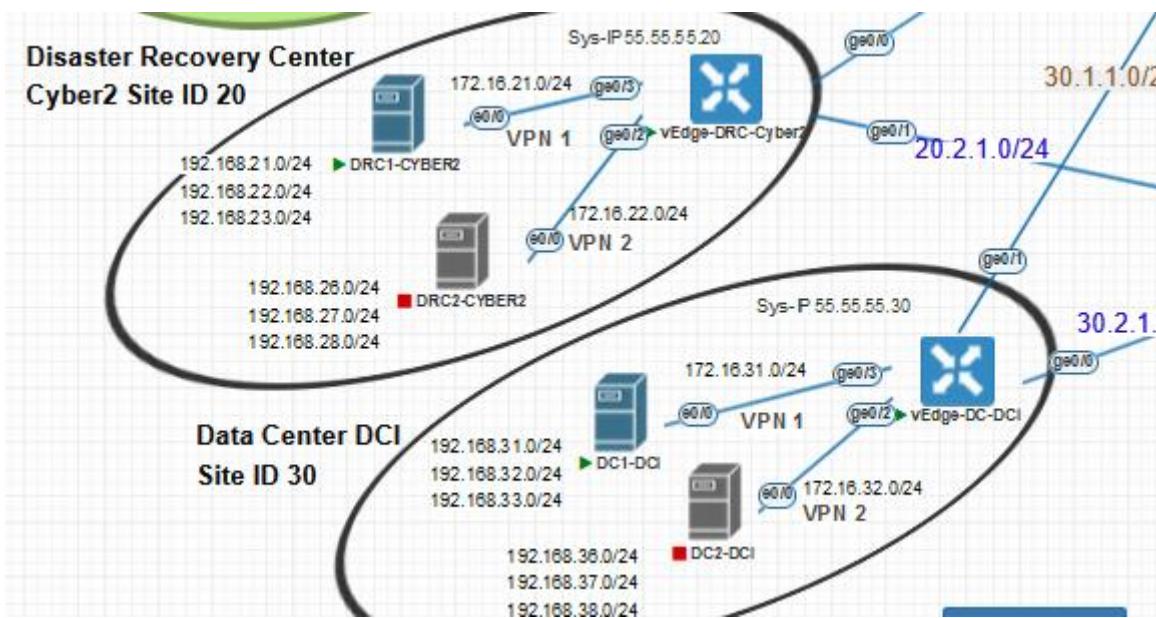
1. Connected Routes
2. Static Routes
3. Dynamic Routes (OSPF, BGP, EIGRP, dsb)

The most important thing is that the vEdge has interconnection to each prefix behind it (LAN Facing), whether using Static or Dynamic Routes. So, for example, when VPN 5 from vEdge1 wants to send a packet to VPN 5 on vEdge 2, the packet does not land on the 'WAN Facing/VPN 0' interface but directly to the 'LAN Facing/VPN 5' interface of vEdge 2.

In essence, the SD-WAN solution takes the routes behind the vEdge and sends them securely via OMP so that other vEdges can learn them. These are not routed via OMP; OMP is simply the control plane that forwards information from one side to the other to allow vEdges to learn the advertised routes.

However, remember that the control plane is not directly connected between vEdge to vEdge. The control plane connects vEdge to vSmart, then vSmart to vEdge (vEdge > vSmart > vEdge). **vSmart holds all the advertised routes for each Service VPN and forwards them to the target Service VPN of each vEdge.**

Scenario:



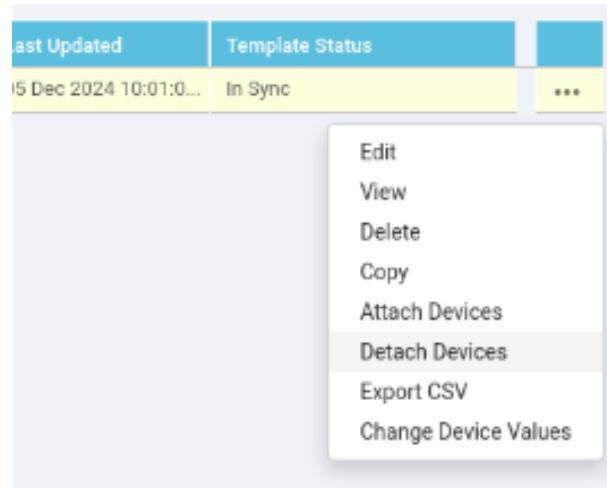
We will use a scenario between vEdge-DRC-Cyber2 (via CLI) and vEdge-DC-DCI (via Templates) for VPN1

VPN 1 : Connected Routes (CLI and Templates)

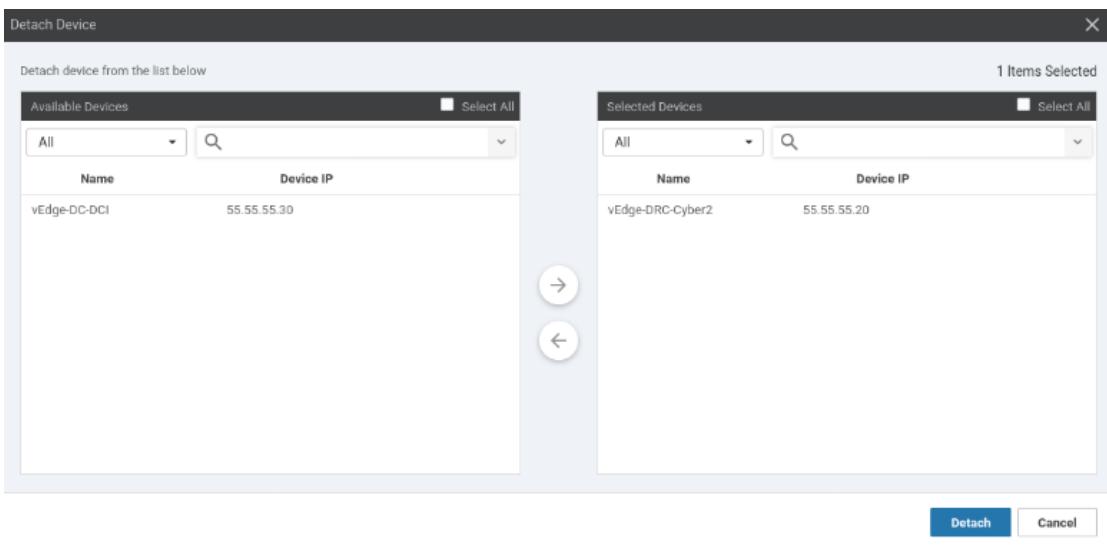
// via CLI

1. [vEdge-DRC-Cyber2]

Since this device was previously part of a Device Template, we can no longer configure it via CLI. First, detach the device. Go to Configuration > Template, then select the '...' on the far right.



Move the device to the right, then ‘Detach.’



This will return vEdge-DRC-Cyber2 to CLI mode.

		Status	Message	Hostname	Device Type	System I	
▼		Success	Done - Device config mode - CLI	vEdge-DRC-Cyber2	WAN Edge	55.55.55	
[6-Dec-2024 10:17:40 WIB]		Changing device configuration mode to CLI					
[6-Dec-2024 10:17:42 WIB]		Changed configuration mode to CLI					

2. Create VPN 1 on vEdge-DRC-Cyber2, and enter the IP address as per the topology:

```
vpn 1
interface ge0/3
ip address 172.16.21.1/24
no shutdown
!
commit
!
```

3. Setup the ‘DRC1-CYBER2’ device (LAN Facing Device).

```
DRC1-CYBER2(config)#do show ip int br | exclude unassign
Interface          IP-Address      OK? Method Status          Protocol
Ethernet0/0        172.16.21.10   YES manual up           up
Loopback1          192.168.21.10  YES manual up           up
Loopback2          192.168.22.10  YES manual up           up
Loopback3          192.168.23.10  YES manual up           up

DRC1-CYBER2(config)#ip route 0.0.0.0 0.0.0.0 172.16.21.1
```

4. As a result, directly connected routes (172.16.21.0/24) will be automatically advertised by vEdge within VPN 1. We can see on show omp peers, there are 2 routes sent to vSmart.

vEdge-DRC-Cyber2# show omp peers							
		DOMAIN ID	OVERLAY ID	SITE ID	STATE	UPTIME	R/I/S
55.55.10.3	vsmart	1	1	10	up	0:00:28:10	0/0/2
55.55.10.4	vsmart	1	1	10	up	0:00:28:10	0/0/2

5. From the vSmart perspective, they receive routes from vEdge 55.55.55.20 and also send 2 routes to other vSmart devices.

vSmart1# show omp peers							
		DOMAIN ID	OVERLAY ID	SITE ID	STATE	UPTIME	R/I/S
55.55.10.4	vsmart	1	1	10	up	0:00:44:01	2/0/2
55.55.55.20	vedge	1	1	20	up	0:00:30:10	2/0/0
55.55.55.30	vedge	1	1	30	up	0:00:41:52	0/0/0

// via Templates

1. [vEdge-DC-DCI]

Here, we will add a new feature template for the interface that connects to VPN 1 (LAN Facing). Go to Configuration > Template, then ‘Add Template’ for VPN 1.

Feature Template	VPN
Device Type	vEdge Cloud
Template Name	vEdge_VPN1_Template
Description	vEdge_VPN1_Template
Basic Configuration	DNS Advertise OMP IPv4 Route IPv6 Route Service GRE Route
BASIC CONFIGURATION	
VPN	1
Name	vEdge_VPN1_Template
Enhance ECMP Keying	Off
Enable TCP Optimization	Off

2. In the 'Advertise OMP' section, select On for 'Connected (IPv4),' then save.

Protocol	Status
BGP (IPv4)	<input type="radio"/> On <input checked="" type="radio"/> Off
Static (IPv4)	<input type="radio"/> On <input checked="" type="radio"/> Off
Connected (IPv4)	<input checked="" type="radio"/> On <input type="radio"/> Off
OSPF External	<input type="radio"/> On <input checked="" type="radio"/> Off
EIGRP	<input type="radio"/> On <input checked="" type="radio"/> Off
LISP	<input type="radio"/> On <input checked="" type="radio"/> Off
ISIS	<input type="radio"/> On <input checked="" type="radio"/> Off

3. Create another feature template for the VPN 1 Interface. Here, we use device-specific options for flexibility.

Device Type	vEdge Cloud						
Template Name	vEdge_VPN1_Int_Template						
Description	vEdge_VPN1_Int_Template						
Basic Configuration	Tunnel	NAT	VRRP	ACL/QoS	ARP	802.1X	Advanced
BASIC CONFIGURATION							
Shutdown	<input type="radio"/> Yes <input checked="" type="radio"/> No						
Interface Name	[vpn1_if_name]						
Description	[vpn1_if_description]						

4. In 'IPv4,' select static, then create device-specific variables as follows, and save.

Basic Configuration	Tunnel	NAT	VRRP	ACL/QoS	ARP	802.1X	Advanced
IPv4 IPv6							
<input type="radio"/> Dynamic <input checked="" type="radio"/> Static							
IPv4 Address	[vpn1_if_ipv4_address]						
Secondary IP Address (Maximum: 4)	<input type="button" value="Add"/>						
DHCP Helper							

5. Add both the created feature templates to the device template.

vEdge_VPN1_Template	vEdge_VPN1_Template	WAN Edge VPN	vEdge Cloud
vEdge_VPN1_Int_Template	vEdge_VPN1_Int_Template	WAN Edge Interface	vEdge Cloud

6. Go to the Device Template ‘vEdge_DC-DRC_Site_Template,’ then edit. In the ‘Service VPN’ section, click ‘+’ to add the feature template.

The screenshot shows a configuration interface for a service VPN. At the top, there's a header 'Service VPN' with a dropdown set to 'Service VPN'. Below it is a large text input field containing the placeholder text 'Add a VPN (any VPN except VPN 0 and VPN 512)'. This indicates that no VPN configurations have been added yet.

7. Add the VPN 1 feature template created earlier, then ‘Update.’

This screenshot shows the 'Service VPN' configuration page again. It displays two dropdown menus: 'VPN' set to 'vEdge_VPN1_Template' and 'VPN Interface' set to 'vEdge_VPN1_Int_Template'. To the right of these dropdowns is a '+' icon followed by the text 'Sub-Templates'. This indicates that the VPN 1 feature template has been successfully added to the service template.

8. For device ‘vEdge-DC-DCI,’ perform ‘Edit Device Template,’ then fill in the information for VPN 1 at the bottom, then ‘Update’ and ‘Next.’

This screenshot shows the 'Edit Device Template' configuration page for a specific device. It includes fields for 'Interface Name(vpn1_if_name)', 'Description(vpn1_if_description)', and 'IPv4 Address(vpn1_if_ipv4_address)'. The values entered are 'ge0/2', 'To VPN1', and '172.16.31.1/24' respectively. At the bottom of the page are three buttons: 'Generate Password', 'Update', and 'Cancel'.

9. Review the configuration that will be pushed to the device. If everything is correct, click ‘Configure Device.’

This screenshot shows a configuration preview screen. It displays the configuration command that will be pushed to the device. The command is as follows:

```

vpn 1
  interface ge0/2
    description "To VPN1"
    ip address 172.16.31.1/24
    no shutdown
  !
  omp
    advertise connected
  !

```

At the bottom of the screen are three buttons: 'Back', 'Configure Devices', and 'Cancel'.

10. As a result, the configuration will be successfully pushed to vEdge-DC-DCI.

```

vEdge-DC-DCI# show run vpn 1
vpn 1
  interface ge0/2
    description "To VPN1"
    ip address 172.16.31.1/24
    no shutdown
  !
  omp
    advertise connected
  !

```

11. Setup ‘DC1-DCI’ device

```
DC1-DCI(config)#do show ip int br | exclude unassi
Interface          IP-Address      OK? Method Status          Protocol
Ethernet0/0        172.16.31.10   YES manual up           up
Loopback1          192.168.31.10  YES manual up           up
Loopback2          192.168.32.10  YES manual up           up
Loopback3          192.168.33.10  YES manual up           up

DC1-DCI(config)#ip route 0.0.0.0 0.0.0.0 172.16.31.1
```

// Verification VPN 1 Interconnection

1. #show omp peers

Both devices receive, install, and send 2 routes.

```
vEdge-DC-DCI# show omp peers
R -> routes received
I -> routes installed
S -> routes sent

PEER          DOMAIN ID    OVERLAY ID    SITE ID    STATE    UPTIME          R/I/S
-----+-----+-----+-----+-----+-----+-----+-----+
55.55.10.3    vsmart  1       1       10       up       0:01:24:19    2/2/2

vEdge-DRC-Cyber2# show omp peers
R -> routes received
I -> routes installed
S -> routes sent

PEER          DOMAIN ID    OVERLAY ID    SITE ID    STATE    UPTIME          R/I/S
-----+-----+-----+-----+-----+-----+-----+-----+
55.55.10.3    vsmart  1       1       10       up       0:01:13:26    2/2/2
```

2. #show ip route

There are 2 routes learned from VPN 1 via OMP, with 2 different transport links

```
vEdge-DRC-Cyber2# show ip route omp
Codes Proto-sub-type:
  IA -> ospf-intra-area, IE -> ospf-inter-area,
  E1 -> ospf-external, 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

VPN  PREFIX          PROTOCOL      PROTOCOL SUB TYPE  NEXTHOP IF NAME  NEXTHOP ADDR  NEXTHOP VPN  TLOC IP          COLOR
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1    172.16.31.0/24  omp          -          -          -          -          -          55.55.55.30  mpls
1    172.16.31.0/24  omp          -          -          -          -          -          55.55.55.30  public-internet

vEdge-DC-DCI# show ip route omp
Codes Proto-sub-type:
  IA -> ospf-intra-area, IE -> ospf-inter-area,
  E1 -> ospf-external, 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

VPN  PREFIX          PROTOCOL      PROTOCOL SUB TYPE  NEXTHOP IF NAME  NEXTHOP ADDR  NEXTHOP VPN  TLOC IP          COLOR
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1    172.16.21.0/24  omp          -          -          -          -          -          55.55.55.20  mpls
1    172.16.21.0/24  omp          -          -          -          -          -          55.55.55.20  public-internet
```

3. # ping & traceroute

```
DC1-DCI#ping 172.16.21.10
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.21.10, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 41/60/72 ms
DC1-DCI#tracer
DC1-DCI#traceroute 172.16.21.10
Type escape sequence to abort.
Tracing the route to 172.16.21.10
VRF info: (vrf in name/id, vrf out name/id)
  1 172.16.31.1 12 msec 11 msec 15 msec
  2 172.16.21.1 48 msec 52 msec 41 msec
  3 172.16.21.10 71 msec 72 msec *
```

```
DRC1-CYBER2#ping 172.16.31.10
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.31.10, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 55/68/72 ms
DRC1-CYBER2#trac
DRC1-CYBER2#traceroute 172.16.31.10
Type escape sequence to abort.
Tracing the route to 172.16.31.10
VRF info: (vrf in name/id, vrf out name/id)
  1 172.16.21.1 20 msec 14 msec 20 msec
  2 172.16.31.1 42 msec 43 msec 31 msec
  3 172.16.31.10 61 msec 69 msec *
```

VPN 1 : Static Routes (CLI and Templates)

// via CLI

1. [vEdge-DRC-Cyber2]

Continuing from the previous step, we will add the static route on the LAN Facing segment, specifically from the DRC1-Cyber2 device.

2. Configure the static route to the DRC1-Cyber2 LAN segment:

```
vpn 1
!
ip route 192.168.21.0/24 172.16.21.10
ip route 192.168.22.0/24 172.16.21.10
ip route 192.168.23.0/24 172.16.21.10
!
commit
```

3. With this configuration, the static routes will automatically be advertised by vEdge within VPN 1. You can see in the show omp peers command, there are 8 routes (1 directly connected, 3 static routes, each with 2 transport links) being sent to vSmart.

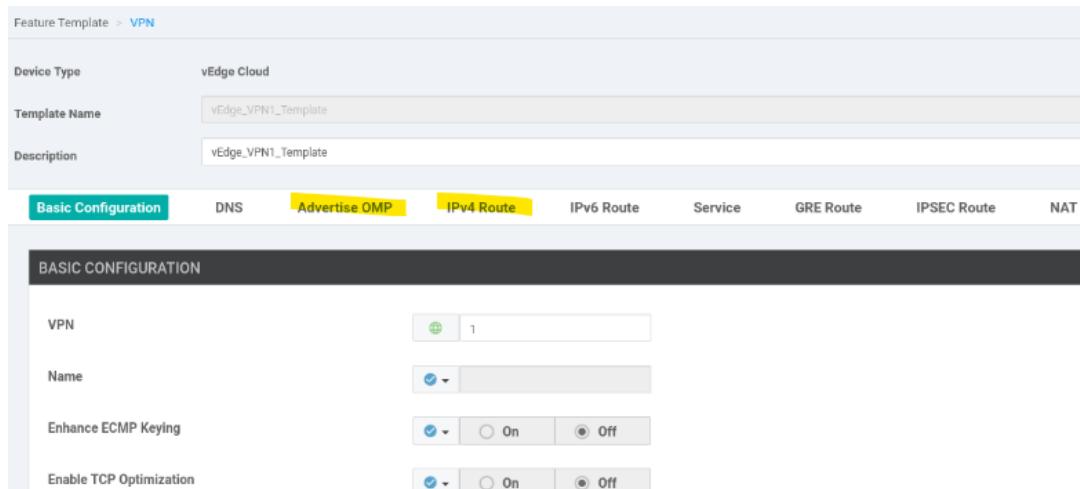
```
vEdge-DRC-Cyber2# show omp peers
R -> routes received
I -> routes installed
S -> routes sent

          DOMAIN      OVERLAY      SITE
PEER      TYPE        ID          ID      ID      STATE      UPTIME      R/I/S
-----+-----+-----+-----+-----+-----+-----+-----+
55.55.10.3    vsmart     1          1       10      up       0:02:36:42   2/2/8
55.55.10.4    vsmart     1          1       10      up       0:02:36:42   2/0/8
```

// via Templates

1. [vEdge-DC-DCI]

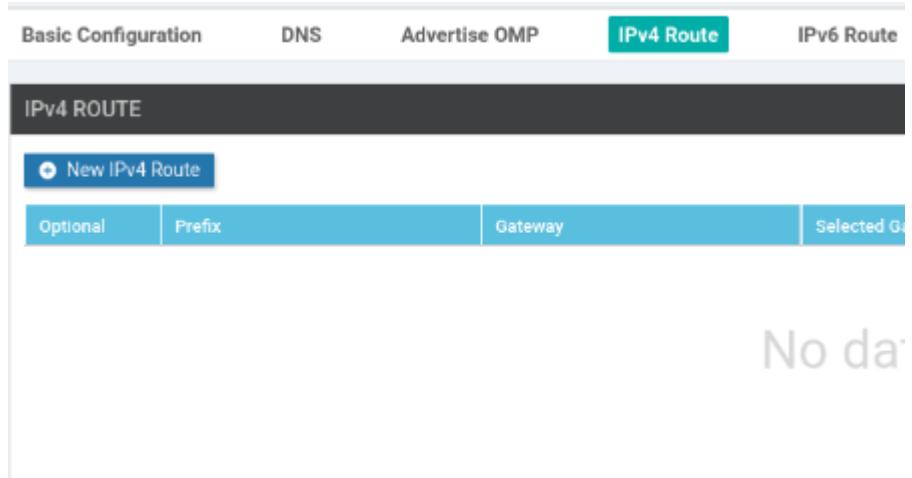
Here, we will only edit the ‘vEdge_VPN1_Template’, specifically in the ‘Advertise OMP’ and ‘IPv4 Route’ sections



2. In ‘Advertise OMP’ section, select ‘On’ for both ‘Connected (IPv4)’ and ‘Static (IPv4)’.



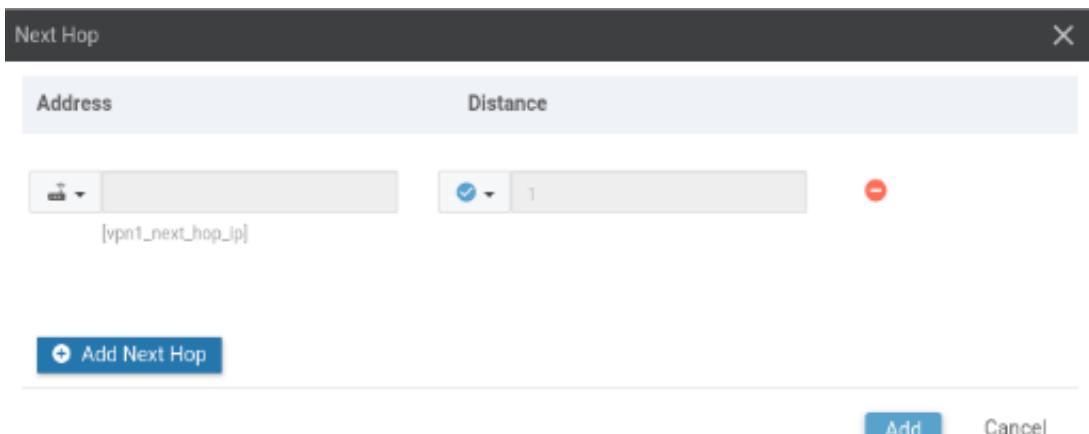
3. Then go to the 'IPv4 Route' section, click 'New IPv4 Route' to add the static IP route.



4. For the prefix, we will use device-specific variables as shown in the image, check 'Mark as Optional Row' to leave it empty if not needed, then click 'Add Next Hop' to add the next-hop address.



5. Choose the device-specific option for the next-hop address, then click 'Add'.



6. With this, the static route is successfully created. Repeat the above steps to create 3 static routes as per the scenario, then click ‘Update’.

Optional	Prefix	Gateway	Selected Gateway Configuration
<input checked="" type="checkbox"/>	[vpn1_ipv4_ip1_prefix]	Next Hop	1
<input checked="" type="checkbox"/>	[vpn1_ipv4_ip2_prefix]	Next Hop	1
<input checked="" type="checkbox"/>	[vpn1_ipv4_ip3_prefix]	Next Hop	1

Update Cancel

7. Add the values or information related to the static routes to ‘vEdge-DC-DCI’. Once done, click ‘Update’, then ‘Next’.

Prefix(vpn1_ipv4_ip1_prefix)	192.168.31.0/24
Prefix(vpn1_ipv4_ip2_prefix)	192.168.32.0/24
Prefix(vpn1_ipv4_ip3_prefix)	192.168.33.0/24
Address(vpn1_next_hop_ip)	172.16.31.10
Address(vpn1_next_hop_ip)	172.16.31.10
Address(vpn1_next_hop_ip)	172.16.31.10

Generate Password

Update Cancel

8. Verify the configuration to be pushed to the device. If correct, click ‘Configure Device’.

```

vpn 1
  interface ge0/2
    description "To VPN1"
    ip address 172.16.31.1/24
    no shutdown
  !
  ip route 192.168.31.0/24 172.16.31.10
  ip route 192.168.32.0/24 172.16.31.10
  ip route 192.168.33.0/24 172.16.31.10
  omp
    advertise connected
    advertise static
  !

```

Back

Configure Devices

9. With that, the configuration is successfully pushed to vEdge-DC-DCI.

```

vEdge-DRC-Cyber2# show run vpn 1
vpn 1
  interface ge0/3
    ip address 172.16.21.1/24
    no shutdown
  !
  ip route 192.168.21.0/24 172.16.21.10
  ip route 192.168.22.0/24 172.16.21.10
  ip route 192.168.23.0/24 172.16.21.10
  !

```

VPN 1 : Verification

// Directly Connected routes

1. #show omp peers

Both devices receive, install, and send 2 routes.

```
vEdge-DC-DCI# show omp peers
R -> routes received
I -> routes installed
S -> routes sent

PEER          DOMAIN ID    OVERLAY ID   SITE ID      STATE    UPTIME           R/I/S
-----+-----+-----+-----+-----+-----+-----+-----+
55.55.10.3    vsmart  1       1        10      up     0:01:24:19    2/2/2
```

```
vEdge-DRC-Cyber2# show omp peers
R -> routes received
I -> routes installed
S -> routes sent

PEER          DOMAIN ID    OVERLAY ID   SITE ID      STATE    UPTIME           R/I/S
-----+-----+-----+-----+-----+-----+-----+-----+
55.55.10.3    vsmart  1       1        10      up     0:01:13:26    2/2/2
```

2. #show ip route omp

There are 2 routes learned from VPN 1 via OMP, with 2 different transport links.

```
vEdge-DRC-Cyber2# show ip route omp
Codes Proto-sub-type:
  IA -> ospf-intra-area, IE -> ospf-inter-area,
  E1 -> ospf-external, E2 -> ospf-external2,
  N1 -> ospf-nssa-external, N2 -> ospf-nssa-external2,
  e -> bgp-external, i -> bgp-internal
Codes Status flags:
  F -> fib, S -> selected, I -> inactive,
  B -> blackhole, R -> recursive

VPN  PREFIX          PROTOCOL      PROTOCOL      NEXTHOP      NEXTHOP      NEXTHOP
      SUB TYPE      SUB TYPE      IF NAME      ADDR        VPN         TLOC IP      COLOR
-----+-----+-----+-----+-----+-----+-----+-----+-----+
1    172.16.31.0/24  omp          -            -            -            -          55.55.55.30  mpls
1    172.16.31.0/24  omp          -            -            -            -          55.55.55.30  public-internet
```

```
vEdge-DC-DCI# show ip route omp
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

VPN  PREFIX          PROTOCOL      PROTOCOL      NEXTHOP      NEXTHOP      NEXTHOP
      SUB TYPE      SUB TYPE      IF NAME      ADDR        VPN         TLOC IP      COLOR
-----+-----+-----+-----+-----+-----+-----+-----+-----+
1    172.16.21.0/24  omp          -            -            -            -          55.55.55.20  mpls
1    172.16.21.0/24  omp          -            -            -            -          55.55.55.20  public-internet
```

3. # ping & traceroute

```
DC1-DCI#ping 172.16.21.10
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.21.10, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 41/60/72 ms
DC1-DCI#tracer
DC1-DCI#traceroute 172.16.21.10
Type escape sequence to abort.
Tracing the route to 172.16.21.10
VRF info: (vrf in name/id, vrf out name/id)
  1 172.16.31.1 12 msec 11 msec 15 msec
  2 172.16.21.1 48 msec 52 msec 41 msec
  3 172.16.21.10 71 msec 72 msec *
```

```
DRC1-CYBER2#ping 172.16.31.10
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.31.10, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 55/68/72 ms
DRC1-CYBER2#traceroute 172.16.31.10
Type escape sequence to abort.
Tracing the route to 172.16.31.10
VRF info: (vrf in name/id, vrf out name/id)
  1 172.16.21.1 20 msec 14 msec 20 msec
  2 172.16.31.1 42 msec 43 msec 31 msec
  3 172.16.31.10 61 msec 69 msec *
```

// Static routes

1. #show omp peers

vEdge-DRC-Cyber2# show omp peers							
R -> routes received I -> routes installed S -> routes sent							
PEER	TYPE	DOMAIN ID	OVERLAY ID	SITE ID	STATE	UPTIME	R/I/S
55.55.10.3	vsmart	1	1	10	up	0:03:52:28	8/8/8

vEdge-DC-DCI# show omp peers							
R -> routes received I -> routes installed S -> routes sent							
PEER	TYPE	DOMAIN ID	OVERLAY ID	SITE ID	STATE	UPTIME	R/I/S
55.55.10.3	vsmart	1	1	10	up	0:04:05:49	8/8/8

2. # show ip route omp

vEdge-DC-DCI# show ip route omp									
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									
VPN	PREFIX	PROTOCOL	PROTOCOL SUB TYPE	NEXTHOP IF NAME	NEXTHOP ADDR	NEXTHOP VPN	TLOC IP	COLOR	
1	172.16.21.0/24	omp	-	-	-	-	55.55.55.20	mpls	
1	172.16.21.0/24	omp	-	-	-	-	55.55.55.20	public-internet	
1	192.168.21.0/24	omp	-	-	-	-	55.55.55.20	mpls	
1	192.168.21.0/24	omp	-	-	-	-	55.55.55.20	public-internet	
1	192.168.22.0/24	omp	-	-	-	-	55.55.55.20	mpls	
1	192.168.22.0/24	omp	-	-	-	-	55.55.55.20	public-internet	
1	192.168.23.0/24	omp	-	-	-	-	55.55.55.20	mpls	
1	192.168.23.0/24	omp	-	-	-	-	55.55.55.20	public-internet	

vEdge-DRC-Cyber2# show ip route omp									
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									
VPN	PREFIX	PROTOCOL	PROTOCOL SUB TYPE	NEXTHOP IF NAME	NEXTHOP ADDR	NEXTHOP VPN	TLOC IP	COLOR	
1	172.16.31.0/24	omp	-	-	-	-	55.55.55.30	mpls	
1	172.16.31.0/24	omp	-	-	-	-	55.55.55.30	public-internet	
1	192.168.31.0/24	omp	-	-	-	-	55.55.55.30	mpls	
1	192.168.31.0/24	omp	-	-	-	-	55.55.55.30	public-internet	
1	192.168.32.0/24	omp	-	-	-	-	55.55.55.30	mpls	
1	192.168.32.0/24	omp	-	-	-	-	55.55.55.30	public-internet	
1	192.168.33.0/24	omp	-	-	-	-	55.55.55.30	mpls	
1	192.168.33.0/24	omp	-	-	-	-	55.55.55.30	public-internet	

3. # ping & traceroute

```
DRC1-CYBER2#ping 192.168.31.10
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.31.10, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 51/63/73 ms
DRC1-CYBER2#ping 192.168.32.10
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.32.10, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 51/69/80 ms
DRC1-CYBER2#ping 192.168.33.10
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.33.10, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 44/66/73 ms
DRC1-CYBER2#traceroute 192.168.33.10
Type escape sequence to abort.
Tracing the route to 192.168.33.10
VRF info: (vrf in name/id, vrf out name/id)
  1 172.16.21.1 20 msec 10 msec 18 msec
  2 172.16.31.1 52 msec 51 msec 51 msec
  3 172.16.31.10 61 msec 63 msec *

DC1-DCI#ping 192.168.21.10
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.21.10, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 61/70/82 ms
DC1-DCI#ping 192.168.22.10
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.22.10, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 59/65/72 ms
DC1-DCI#ping 192.168.23.10
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.23.10, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 63/69/72 ms
DC1-DCI#tracer
DC1-DCI#traceroute 192.168.23.10
Type escape sequence to abort.
Tracing the route to 192.168.23.10
VRF info: (vrf in name/id, vrf out name/id)
  1 172.16.31.1 15 msec 20 msec 20 msec
  2 172.16.21.1 41 msec 58 msec 51 msec
  3 172.16.21.10 72 msec 63 msec *
```

SERVICE VPN: VRRP & DHCP

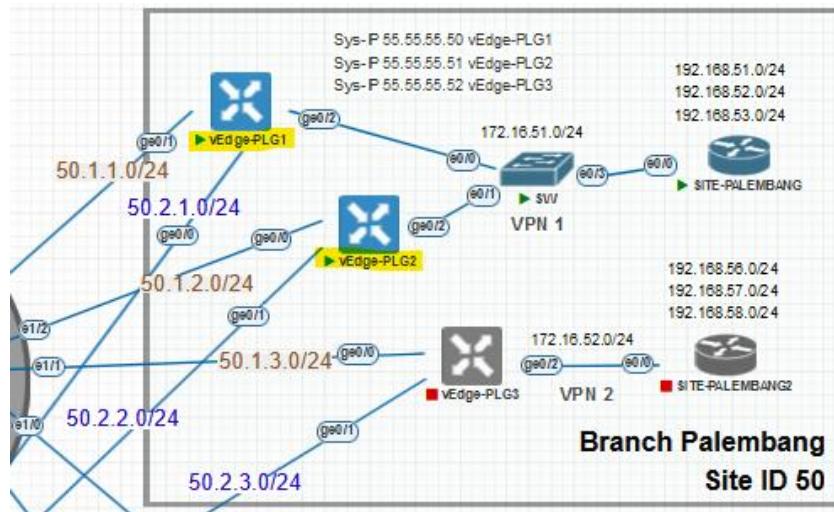
Via CLI and Templates

Scenario:

In this scenario, we will use the Palembang Site (Site 50), with vEdge-PLG1 and vEdge-PLG2. The features that we will apply here include:

1. Virtual Router Redundancy Protocol (VRRP)
2. Dynamic Host Configuration Protocol (DHCP)

The gateway used for each network in Service VPN 1 / LAN Facing is the IP from VRRP, and the interface leading to vEdge from the 'SITE-PALEMBANG' will use DHCP.



vEdge-PLG1 will be configured through CLI, while vEdge-PLG2 will be configured using templates. We will create a new Device Template for vEdge-PLG2, involving the feature templates that have been created, as follows:

- Basic Configuration > System
- VPN > VPN 0 (VPN (Default Route, OSPF), VPN Interface)
- VPN > VPN 512 (VPN, VPN Interface)
- VPN > VPN 1 (VPN, VPN Interface) << we have not yet applying routing protocol
- Other Template > Banner

VPN 1 : VRRP (CLI and Templates)**// via CLI**

1. [vEdge-PLG1]
2. Set up VPN 1 and configure VRRP on interface Ge0/2 (LAN Facing)

```

vpn 1

interface ge0/2
    ip address 172.16.51.1/24
    no shutdown
    vrrp 50
    priority 200
    track-omp           << if there is problem with omp, the priority decreased
    ipv4 172.16.51.25
    !
commit
!

```

// via Templates

1. [vEdge-PLG2]

Copy feature templates for VPN 0 and VPN 1 for the Palembang Site and remove the template configuration used previously for DC & DRC, to ensure that the configuration for the DC & DRC device templates is not disturbed.

vEdge_PLG_VPN0_Int_Ge0/0_Template	vEdge_PLG_VPN0_Int_Ge0/...	WAN Edge Interface
vEdge_PLG_VPN0_Int_Ge0/1_Template	vEdge_PLG_VPN0_Int_Ge0/...	WAN Edge Interface
vEdge_PLG_VPN0_Template	vEdge_PLG_VPN0_Template	WAN Edge VPN
vEdge_PLG_VPN1_Int_Template	vEdge_PLG_VPN1_Int_Tem...	WAN Edge Interface
vEdge_PLG_VPN1_Template	vEdge_PLG_VPN1_Template	WAN Edge VPN

2. Create a new device template for the Palembang Site..

CONFIGURATION | TEMPLATES

Device **Feature**

Device Model	vEdge Cloud
Template Name	vEdge_Palembang_Site_Template
Description	vEdge_Palembang_Site_Template

3. Basic Configuration

Basic Information	Transport & Management VPN	Service VPN	Additional Templates
Basic Information			
System *	vEdge_System_Template		
Logging*	Factory_Default_Logging_Template		
AAA *	Factory_Default_AAA_Template	BFD *	
OMP *	Factory_Default_vEdge_OMP_Template	Security *	

4. Transport & Management VPN

Transport & Management VPN			
VPN 0 *	vEdge_PLG_VPN0_Template		
OSPF	vEdge_VPN0 OSPF_Template		
VPN Interface	vEdge_PLG_VPN0_Int_Ge0/0_Template		
VPN Interface	vEdge_PLG_VPN0_Int_Ge0/1_Template		
VPN 512 *	vEdge_VPN512_Template		
VPN Interface	vEdge_VPN512_Int_eth0_Template		

5. Service VPN

The screenshot shows the 'Service VPN' configuration page. At the top, there's a header with 'Service VPN' and a dropdown. Below it, under 'VPN', a dropdown menu is set to 'vEdge_PLG_VPN1_Template'. Under 'VPN Interface', another dropdown is set to 'vEdge_PLG_VPN1_Int_Template', with a 'Sub-Templates' button next to it.

6. Additional Templates, then 'Create'

The screenshot shows a 'Additional Templates' dialog box. It contains four sections: 'Banner' (set to 'vEdge_Banner_Template'), 'Policy' (with a 'Choose...' dropdown), 'SNMP' (with a 'Choose...' dropdown), and 'Security Policy' (with a 'Choose...' dropdown). At the bottom right are 'Create' and 'Cancel' buttons.

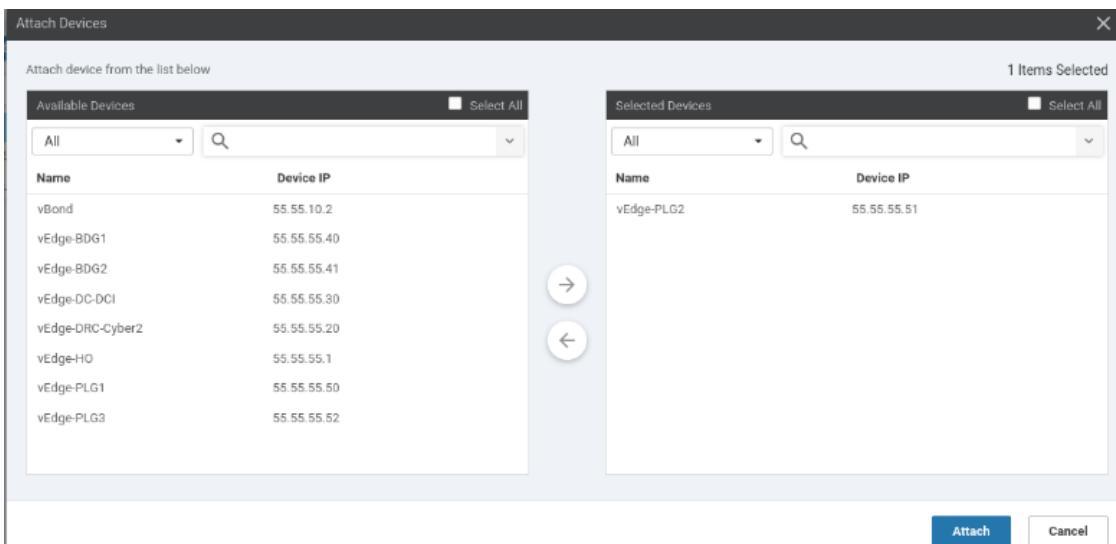
7. Configure VRRP on interface Ge0/2 (LAN Facing) on VPN 1. Go to the feature template for VPN 1 Interface, then go to the 'VRRP' section.

The screenshot shows the 'VRRP' configuration interface. A top bar has 'IPv4' and 'IPv6' tabs, with 'IPv4' selected. Below is a table with columns: 'Optional', 'Group ID', 'Priority', 'Timer', 'Track OMP', and 'Tra'. A message 'No data available' is displayed below the table.

8. Click 'New VRRP', then make all parameters device-specific. We will configure them when attaching to the device, then click 'Add' and 'Update'.

The screenshot shows the 'New VRRP' configuration dialog. It includes fields for 'Group ID' (set to 'vpn1_Ge0-2_vrrp_group'), 'Priority' (set to 'vpn1_Ge0-2_vrrp_priority'), 'Timer (seconds)' (set to '1'), 'Track OMP' (set to 'On'), and 'IP Address' (set to 'vpn1_Ge0-2_vrrp_ipaddress'). A 'Mark as Optional Row' checkbox is checked. At the bottom are 'Add' and 'Cancel' buttons.

9. Attach the device to the device template that has been created.



10. Edit all parameter values according to the topology. For VRRP, we will make vEdge-PLG1 the backup VRRP, so we set its priority to 150. Once all fields are filled, click 'Update'.

Update Device Template

Variable List (Hover over each field for more information)

Chassis Number	55d1e2ba-caa0-4466-bc89-80971bcc383f
System IP	55.55.55.51
Hostname	vEdge-PLG2
Interface Name(vpn1_if_name)	ge0/2
Description(vpn1_if_description)	To VPN 1 Lan facing
IPv4 Address(vpn1_if_ipv4_address)	172.16.51.2/24
Group ID(vpn1_Ge0-2_vrrp_grpid)	50
Priority(vpn1_Ge0-2_vrrp_priority)	150
IP Address(vpn1_Ge0-2_vrrp_vrrp_ipaddress)	172.16.51.254
Address(vpn0_next_hop_ip)	50.2.2.2
Description(vpn0_int_Ge0-1_description)	TO PUBLIC INTERNET
IPv4 Address(vpn0_int_Ge0-1_ipv4_address)	50.2.2.1/24
Color(vpn0_int_Ge0-1_tunnel_color_value)	public-internet
Description(vpn0_int_Ge0-0_description)	To MPLS
IPv4 Address(vpn0_int_Ge0-0_ipv4_address)	50.1.2.1/24
Color(vpn0_int_Ge0-0_tunnel_color_value)	mpls

Buttons: Generate Password, Update (blue), Cancel

11. Verify the configuration preview before pushing. If everything is correct, click 'Configure Device'.

```

vpn 1
interface ge0/2
description "To VPN 1 Lan Facing"
ip address 172.16.51.2/24
no shutdown
vrrp 50
priority 150
track-omp
ipv4 172.16.51.254
!
!
!
```

12. The configuration is successfully pushed to vEdge-PLG2.

```

vEdge-PLG2# show run vpn 1
vpn 1
interface ge0/2
description "To VPN 1 Lan facing"
ip address 172.16.51.2/24
no shutdown
vrrp 50
priority 150
track-omp
ipv4 172.16.51.254
!
```

VPN 1 : DHCP (CLI and Templates)

// via CLI

- [vEdge-PLG1]**
- Configure DHCP on VPN 1 interface Ge0/2 (LAN Facing)

```

vpn 1
interface ge0/2
dhcp-server
address-pool 172.16.51.0/24
exclude 172.16.51.1 172.16.51.2 172.16.51.254
options
default-gateway 172.16.51.254
dns-servers 8.8.8.8
!
```

// via Templates

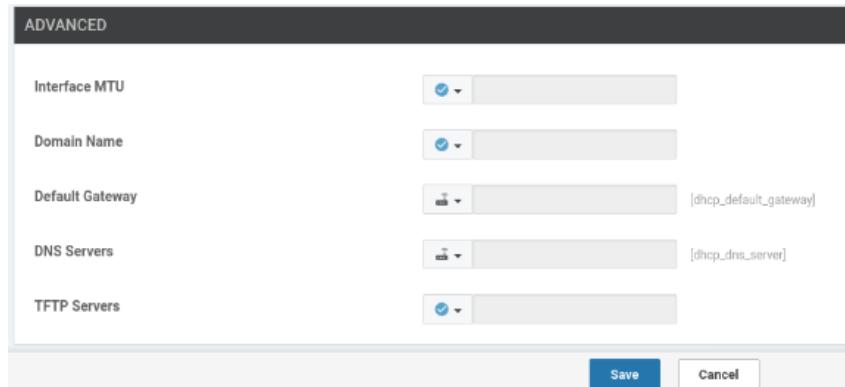
1. [vEdge-PLG2]
2. Add a new feature template, Other Template > DHCP



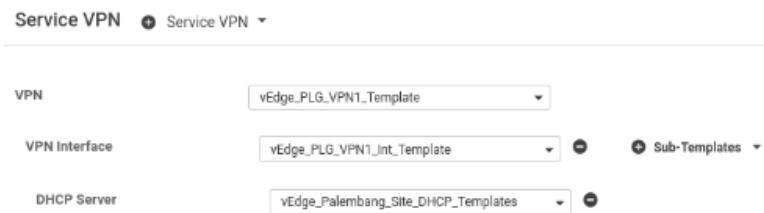
3. Create template name and choose device-specific option for address pool and exclusions

Device	Feature
Feature Template > Add Template > DHCP Server	
Template Name	vEdge_PLG_DHCP_Templates
Description	vEdge_PLG_DHCP_Templates
Basic Configuration	Static Lease DHCP Options Advanced
BASIC CONFIGURATION	
Address Pool	<input type="text"/> [dhcp_address_pool]
Exclude Addresses	<input type="text"/> [dhcp-address_exclude]
Maximum Leases	<input type="text"/>
Lease Time (seconds)	<input type="text"/> 86400
Offer Time (seconds)	<input type="text"/> 600
Administrative State	<input type="button"/> up

4. In the ‘Advanced’ section, choose device-specific options for ‘Default gateway’ and ‘DNS Server’ which will be configured later, then save.



5. Apply DHCP to the Palembang Site device template, in the ‘Service VPN’ section, specifically in the VPN 1 Interface template. Add ‘Sub-templates’ and input the DHCP-Server template that has been created, then click ‘Update’.

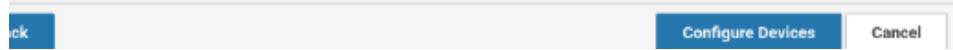


6. On vEdge-PLG2, ‘Edit Device Template’ then fill in the necessary DHCP information, finally click ‘Update’ and ‘Next’.



7. Verify the configuration before pushing to the device, and if everything looks good, click ‘Configure Devices’.

```
!
dhcp-server
  address-pool 172.16.51.0/24
  exclude      172.16.51.1 172.16.51.2 172.16.51.254
  offer-time   600
  lease-time   86400
  admin-state  up
  options
    default-gateway 172.16.51.254
    dns-servers     8.8.8.8
!
!
```



8. The configuration is successfully pushed to 'vEdge-PLG2'.

```

vpn 1
  interface ge0/2
    description "To VPN 1 Lan facing"
    ip address 172.16.51.2/24
    no shutdown
  vrrp 50
    priority 150
    track-omp
    ipv4 172.16.51.254
  !
  dhcp-server
    address-pool 172.16.51.0/24
    exclude 172.16.51.1 172.16.51.2 172.16.51.254
    offer-time 600
    lease-time 86400
    admin-state up
    options
      default-gateway 172.16.51.254
      dns-servers 8.8.8.8
  !
  !
  !

```

VPN 1 : Verification

```
# show vrrp
```

```

vEdge-PLG1# show vrrp
vrrp vpn 1
  interfaces ge0/2
  groups 50
    virtual-ip          172.16.51.254
    virtual-mac         00:00:5e:00:01:32
    priority            200
    vrrp-state          master
    omp-state           up
    advertisement-timer 1
    master-down-timer   3
    last-state-change-time 2024-12-07T09:08:10+00:00

```

```

vEdge-PLG2# show vrrp
vrrp vpn 1
  interfaces ge0/2
  groups 50
    virtual-ip          172.16.51.254
    virtual-mac         00:00:5e:00:01:32
    priority            150
    vrrp-state          backup
    omp-state           up
    advertisement-timer 1
    master-down-timer   3
    last-state-change-time 2024-12-07T02:16:53+00:00

```

DHCP Request from 'SITE-PALEMBANG' device

```
SITE-PALEMBANG(config)#int e0/0
SITE-PALEMBANG(config-if)#ip add
SITE-PALEMBANG(config-if)#ip address dhcp
SITE-PALEMBANG(config-if)#no shut
SITE-PALEMBANG(config-if)#do show ip
*Dec 7 03:02:55.728: %LINK-3-UPDOWN: Interface Ethernet0/0, changed state to up
*Dec 7 03:02:56.730: %LINEPROTO-5-UPDOWN: Line protocol on Interface Ethernet0/0, changed state to up
SITE-PALEMBANG(config-if)#do show ip int br
*Dec 7 03:03:01.432: %DHCP-6-ADDRESS_ASSIGN: Interface Ethernet0/0 assigned DHCP address 172.16.51.171, mask 255.255.255.0,
SITE-PALEMBANG(config-if)#do show ip int br
Interface          IP-Address      OK? Method Status           Protocol
Ethernet0/0        172.16.51.171  YES DHCP   up                up
Ethernet0/1        unassigned     YES unset  administratively down
Ethernet0/2        unassigned     YES DHCP   administratively down down
Ethernet0/3        unassigned     YES unset  administratively down down
Ethernet1/0        unassigned     YES unset  administratively down down
Ethernet1/1        unassigned     YES unset  administratively down down
Ethernet1/2        unassigned     YES unset  administratively down down
Ethernet1/3        unassigned     YES unset  administratively down down
Loopback1          192.168.51.10  YES manual up               up
Loopback2          192.168.52.10  YES manual up               up
Loopback3          192.168.53.10  YES manual up               up

vEdge-PLG1# show dhcp server
dhcp server vpn 1 ge0/2
  bindings aa:bb:cc:00:0b:00
    client-ip          172.16.51.171
    lease-time         1:00:00:00
    lease-time-remaining 0:23:58:03
    static-binding     false
    host-name          SITE-PALEMBANG
```

VRRP Test

```
SITE-PALEMBANG#traceroute 192.168.21.10
Type escape sequence to abort.
Tracing the route to 192.168.21.10
VRF info: (vrf in name/id, vrf out name/id)
  1 172.16.51.1 20 msec 21 msec 20 msec
  2 172.16.21.1 53 msec 42 msec 51 msec
  3 *
      172.16.21.10 48 msec *
```

vEdge-PLG1 Master

```
vEdge-PLG1(config-interface-ge0/2)# int ge0/2
vEdge-PLG1(config-interface-ge0/2)# shutdown
vEdge-PLG1(config-interface-ge0/2)# commit
```

Shutdown Master VRRP

```
vrrp vpn 1
  interfaces ge0/2
  groups 50
    virtual-ip          172.16.51.254
    virtual-mac         00:00:5e:00:01:32
    priority            150
    vrrp-state          master
    oom-state           up
    advertisement-timer 1
    master-down-timer   3
    last-state-change-time 2024-12-07T03:07:59+00:00
```

vEdge-PLG2 takeover Master role

```
SITE-PALEMBANG#traceroute 192.168.21.10
Type escape sequence to abort.
Tracing the route to 192.168.21.10
VRF info: (vrf in name/id, vrf out name/id)
  1 172.16.51.2 21 msec 21 msec 20 msec
  2 172.16.21.1 42 msec 51 msec 51 msec
  3 172.16.21.10 71 msec * 69 msec
```

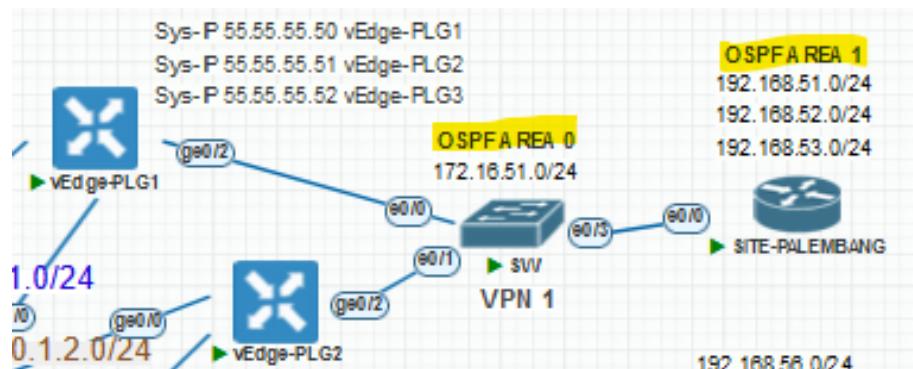
vEdge-PLG2 Master

SERVICE VPN: OSPF CONFIGURATION

Via CLI and Templates

Scenario:

In the interconnection of Service VPN 1 or LAN Facing, we will connect each segment using OSPF, building neighbor peering between the vEdge-PLG device and SITE-PALEMBANG. Here, we will use the **default OSPF network type**, which is **Broadcast**, so there will be **Type 2 LSA** or Network LSA that function as DR/BDR packets both during the election process and in operation.



The segments connected to the switches between the three devices mentioned above are included in OSPF area 0, while the loopback address or the segment behind the 'SITE-PALEMBANG' device is included in OSPF area 1. We will configure the connectivity of Service VPN 1 with OSPF, where the vEdge-PLG1 device will configure it via CLI, and vEdge-PLG2 will do it through vManage/Templates.

Setup LAN Facing Device

1. [SITE-PALEMBANG]
2. Configure OSPF to establish a neighborship with vEdge so that the prefixes on SITE-PALEMBANG will be advertised to VPN 1

```
router ospf 1
router-id 50.50.50.100
```

```

network 0.0.0.0 255.255.255.255 area 0
!
Interface range lo1-3
ip ospf network point-to-point
!
```

VPN 1 : OSPF PEERING VIA CLI

1. [vEdge-PLG1]
2. Configure OSPF with default-information originate to redistribute the default route to the public internet, then use redistribute omp to allow 'SITE-PALEMBANG' to receive prefixes from OMP from other sites. Define OSPF area 0 on interface ge0/2 (LAN Facing). Apply a cost of 5 to make vEdge-PLG1 the primary link for OSPF (according to its VRRP role).

```

vpn 1
router
ospf
router-id 55.55.55.50
default-information originate
redistribute omp
area 0
interface ge0/2
cost 5
exit
exit
!
commit
!
```

VPN 1 : OSPF PEERING VIA TEMPLATES

1. [vEdge-PLG2]

2. Create a new feature template for OSPF to be applied to Service VPN 1.



3. Specify the template name and description, then change the Router ID to device specific.

The screenshot shows the 'Add Template' configuration page for OSPF. The 'Device Type' is set to 'vEdge Cloud'. The 'Template Name' is 'vEdge_PLG_VPN1_OSPF_Template' and the 'Description' is 'vEdge_PLG_VPN1_OSPF_Template'. The 'Basic Configuration' tab is selected, showing the following fields:

- Router ID:** A dropdown menu with a value of 'ospf_router_id'.
- Distance for External Routes:** A dropdown menu with a value of '110'.
- Distance for Inter-Area Routes:** A dropdown menu with a value of '110'.
- Distance for Intra-Area Routes:** A dropdown menu with a value of '110'.

4. In the 'Redistribute' section, add redistribution for OMP, select device specific.

The screenshot shows the 'Redistribute' configuration page. The 'Redistribute' tab is selected. A 'New Redistribute' button is visible. The 'Protocol' field has a dropdown menu with a value of 'ospf_redistribute_protocol'. The 'Route Policy' field has a dropdown menu with a value of 'ospf_route_policy'. There are 'Add' and 'Cancel' buttons at the bottom right.

5. In the 'Area' section, add the area number as device specific, then click 'Add Interface'.

6. In the 'Interface' menu, configure the interface name and cost as device specific.

7. You can also configure network type and OSPF authentication by clicking on 'Advanced Options'.

8. Once done, click 'Add' in the Interface menu, then 'Add' again in the Area section.

Optional	Number	Area Type
<input checked="" type="checkbox"/>	<input type="text"/> [ospf_area_a_n...]	<input checked="" type="checkbox"/>

9. In the 'Advanced' section, enable 'Originate' for the default-information originate feature, then click 'Save'.

Reference Bandwidth (Mbps)	<input checked="" type="checkbox"/> 100
RFC 1583 Compatible	<input checked="" type="checkbox"/> On <input type="radio"/> Off
Originate	<input checked="" type="checkbox"/> On <input type="radio"/> Off
Always	<input checked="" type="checkbox"/> On <input type="radio"/> Off
Default Metric	<input checked="" type="checkbox"/>
Metric Type	<input checked="" type="checkbox"/>
SPF Calculation Delay (milliseconds)	<input checked="" type="checkbox"/> 200

Save **Cancel**

10. Add the OSPF feature template that has been created to VPN 1 in the Palembang Site device template, then click 'Update'.

VPN	vEdge_PLG_VPN1_Template
OSPF	vEdge_PLG_VPN1_OSPF_Template
VPN Interface	vEdge_PLG_VPN1_Int_Template
DHCP Server	vEdge_Palembang_Site_DHCP_Templates

11. Edit vEdge-PLG2 with the OSPF information to be applied, click 'Update' and 'Next'.

Router ID(ospf_router_id)	55.55.55.51
Interface Name(ospf_area_0_if_name)	ge0/0
Router ID(ospf_router_id)	55.55.55.51
Protocol(ospf_redistribute_protocol)	omp
Area Number(ospf_area_a_num)	0
Interface Name(ospf_if_name)	ge0/2
Interface Cost(ospf_cost)	10

Generate Password **Update**

12. Verify the configuration before pushing it to vEdge-PLG2, and if everything is correct, click 'Configure Devices'.

```
vpn 1
router
ospf
  router-id 55.55.55.51
  default-information originate
  timers spf 200 1000 10000
  redistribute omp
  area 0
    interface ge0/2
    cost 10
  exit
exit
!
```

Back**Configure Devices**

13. The device template will successfully be pushed to vEdge-PLG2.

```
vEdge-PLG2# show run vpn 1
vpn 1
  router
    ospf
      router-id 55.55.55.51
      default-information originate
      timers spf 200 1000 10000
      redistribute omp
      area 0
        interface ge0/2
        cost 10
      exit
    exit
!
```

VPN 1 : OSPF Verification

// show ospf neighbor

```
vEdge-PLG1# show ospf neighbor
DBsmL -> Database Summary List
RqstL -> Link State Request List
RXmtL -> Link State Retransmission List
      SOURCE
VPN   IP ADDRESS     INTERFACE     ROUTER ID     STATE
-----
0     50.1.1.2         ge0/1        2.2.2.4       full
1     172.16.51.2       ge0/2        55.55.55.51   full
1     172.16.51.171     ge0/2        50.50.50.100  full
```

```
vEdge-PLG2# show ospf neighbor
DBsmL -> Database Summary List
RqstL -> Link State Request List
RXmtL -> Link State Retransmission List
      SOURCE
VPN   IP ADDRESS     INTERFACE     ROUTER ID     STATE
-----
0     50.1.2.2         ge0/0        2.2.2.4       full
1     172.16.51.1       ge0/2        55.55.55.50   full
1     172.16.51.171     ge0/2        50.50.50.100  full
```

```
SITE-PALEMBANG#show ip ospf neighbor
Neighbor ID      Pri  State          Dead Time    Address      Interface
55.55.55.50      1    FULL/BDR      00:00:30    172.16.51.1  Ethernet0/0
55.55.55.51      1    FULL/DR       00:00:37    172.16.51.2  Ethernet0/0
```

// show ip route on SITE-PALEMBANG

What is special about SD-WAN is its ability to segment the network without concern for what happens in the Transport VPN. SITE-PALEMBANG receives routes from VPN 1.

```
SITE-PALEMBANG#show ip route ospf
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route, H - NHRP, l - LISPs
      a - application route
      + - replicated route, % - next hop override, p - overrides from PfR

Gateway of last resort is 172.16.51.254 to network 0.0.0.0

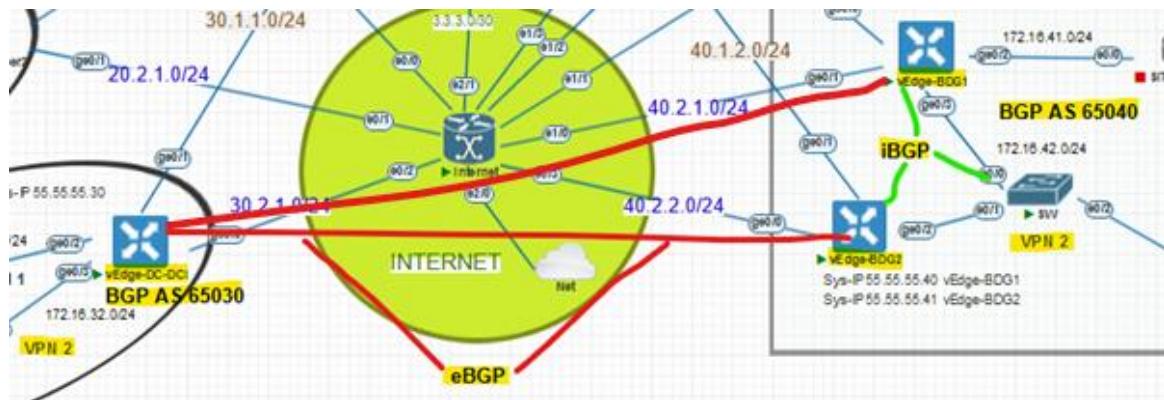
      172.16.0.0/16 is variably subnetted, 4 subnets, 2 masks
0 E2    172.16.21.0/24 [110/16777214] via 172.16.51.2, 00:11:42, Ethernet0/0
                  [110/16777214] via 172.16.51.1, 00:11:42, Ethernet0/0
0 E2    172.16.31.0/24 [110/16777214] via 172.16.51.2, 00:05:05, Ethernet0/0
                  [110/16777214] via 172.16.51.1, 00:05:04, Ethernet0/0
0 E2    192.168.21.0/24 [110/16777214] via 172.16.51.2, 00:11:42, Ethernet0/0
                  [110/16777214] via 172.16.51.1, 00:11:42, Ethernet0/0
0 E2    192.168.22.0/24 [110/16777214] via 172.16.51.2, 00:11:42, Ethernet0/0
                  [110/16777214] via 172.16.51.1, 00:11:42, Ethernet0/0
0 E2    192.168.23.0/24 [110/16777214] via 172.16.51.2, 00:11:42, Ethernet0/0
                  [110/16777214] via 172.16.51.1, 00:11:42, Ethernet0/0
0 E2    192.168.31.0/24 [110/16777214] via 172.16.51.2, 00:05:05, Ethernet0/0
                  [110/16777214] via 172.16.51.1, 00:05:04, Ethernet0/0
0 E2    192.168.32.0/24 [110/16777214] via 172.16.51.2, 00:05:05, Ethernet0/0
                  [110/16777214] via 172.16.51.1, 00:05:04, Ethernet0/0
0 E2    192.168.33.0/24 [110/16777214] via 172.16.51.2, 00:05:05, Ethernet0/0
                  [110/16777214] via 172.16.51.1, 00:05:04, Ethernet0/0
```

```
// ping segment vEdge-DC-DCI and vEdge-DRC-Cyber2 from SITE-PALEMBANG
Success rate is 100 percent (5/5), round-trip min/avg/max = 41/54/62 ms
SITE-PALEMBANG#ping 192.168.21.10
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.21.10, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 38/54/71 ms
SITE-PALEMBANG#ping 192.168.22.10
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.22.10, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 42/51/61 ms
SITE-PALEMBANG#ping 192.168.23.10
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.23.10, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 55/59/64 ms
SITE-PALEMBANG#ping 192.168.31.10
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.31.10, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 51/63/72 ms
SITE-PALEMBANG#ping 192.168.32.10
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.32.10, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 56/64/72 ms
SITE-PALEMBANG#ping 192.168.33.10
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.33.10, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 55/62/72 ms
```

SERVICE VPN: BGP CONFIGURATION

Via CLI and Templates

Scenario:



In this scenario, for configuring BGP, we will use VPN 2. We will establish iBGP neighbor peering within the Bandung Site and eBGP neighbor peering between the Bandung Site and the DC-DCI Site.

vEdge-BDG1 will be configured through CLI, while vEdge-BDG2 and vEdge-DC-DCI will be configured using templates. For the vEdge-DC-DCI device, we will only need to add the BGP configuration and a few additional settings. Meanwhile, for vEdge-BDG2, a new Device Template will be created by incorporating the feature templates that have already been created, as described below.

- Basic Configuration > System
- VPN > VPN 0 (VPN (Default Route, OSPF), VPN Interface)
- VPN > VPN 512 (VPN, VPN Interface)
- VPN > VPN 1 (VPN, VPN Interface, OSPF) << we have not yet apply BGP routing
- Other Template > Banner

To provide a brief understanding, BGP has two types: Internal BGP (iBGP) and External BGP (eBGP). The difference between them lies in the scope of their connection. iBGP

operates within the same autonomous system (AS), while eBGP connects different autonomous systems. Other differences can be observed in the diagram below:

What's the deal with iBGP versus eBGP?	
eBGP	iBGP
Between Autonomous Systems	Across an Autonomous System
Time to live – 1	Time to live – 255
Modifies AS path and Next Hop attributes	Does not modify AS path and Next Hop attributes
Does not use Local Preference attribute	Does use Local Preference attribute
Administrative Distance – 20	Administrative Distance – 200
Advertises eBGP and iBGP learned routes to eBGP peer	Advertises eBGP learned routes to iBGP peer
Does not require full mesh of relationships	Does require full mesh of relationships

BGP is a type of routing protocol that uses path attributes to determine its routes, differing from OSPF and EIGRP, which rely on cost or metrics. In BGP, we can influence the paths taken by each prefix propagated or advertised by BGP routers. Therefore, in this BGP section, we will also configure the protocol to influence both inbound and outbound traffic paths used to reach or direct specific traffic.

Setup LAN Facing Device

// SITE-BANDUNG2

- Configure the **SITE-BANDUNG2** device using **OSPF** for reachability within the Bandung Site and establish an **iBGP peering** with the vEdge device. Adjust the **Administrative Distance (AD)** so that it is higher than iBGP, ensuring that the iBGP protocol is selected as the best route. Advertise all **SITE-BANDUNG loopback prefixes** as follows:

```
interface Loopback0
ip address 40.40.40.100 255.255.255.0
!
interface Loopback1
ip address 192.168.46.10 255.255.255.0
```

```
ip ospf network point-to-point
!
interface Loopback2
ip address 192.168.47.10 255.255.255.0
ip ospf network point-to-point
!
interface Loopback3
ip address 192.168.48.10 255.255.255.0
ip ospf network point-to-point
!
router ospf 1
router-id 40.40.40.100
network 0.0.0.0 255.255.255.255 area 0
distance 210
!
router bgp 65040
bgp router-id 40.40.40.100
bgp log-neighbor-changes
no bgp default ipv4-unicast
neighbor 40.40.40.1 remote-as 65040
neighbor 40.40.40.1 update-source Loopback0
neighbor 40.40.40.2 remote-as 65040
neighbor 40.40.40.2 update-source Loopback0
!
address-family ipv4
network 192.168.46.0 mask 255.255.255.0
network 192.168.47.0 mask 255.255.255.0
network 192.168.48.0 mask 255.255.255.0
neighbor 40.40.40.1 activate
neighbor 40.40.40.1 soft-reconfiguration inbound
```

```
neighbor 40.40.40.2 activate  
neighbor 40.40.40.2 soft-reconfiguration inbound  
exit-address-family  
!
```

// DC2-DCI

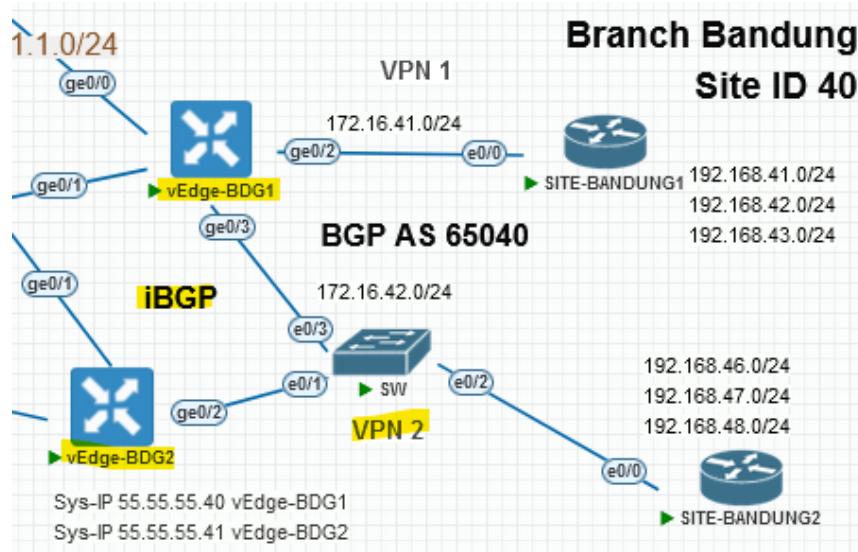
1. Configure the **DC2-DCI** device using **OSPF** for reachability within the DC2-DCI environment and establish an **iBGP peering** with the vEdge device. Adjust the **Administrative Distance (AD)** to prioritize iBGP over OSPF and advertise all **DC2-DCI loopback prefixes** as follows:

```
interface Loopback0  
ip address 30.30.30.100 255.255.255.0  
!  
interface Loopback1  
ip address 192.168.36.10 255.255.255.0  
ip ospf network point-to-point  
!  
interface Loopback2  
ip address 192.168.37.10 255.255.255.0  
ip ospf network point-to-point  
!  
interface Loopback3  
ip address 192.168.38.10 255.255.255.0  
ip ospf network point-to-point  
!  
router ospf 1  
router-id 30.30.30.100  
network 0.0.0.0 255.255.255.255 area 0  
distance 210  
!
```

```
router bgp 65030
bgp router-id 30.30.30.100
bgp log-neighbor-changes
no bgp default ipv4-unicast
neighbor 30.30.30.1 remote-as 65030
neighbor 30.30.30.1 update-source Loopback0
!
address-family ipv4
network 192.168.36.0 mask 255.255.255.0
network 192.168.37.0 mask 255.255.255.0
network 192.168.38.0 mask 255.255.255.0
neighbor 30.30.30.1 activate
neighbor 30.30.30.1 soft-reconfiguration inbound
exit-address-family
!
```

VPN 2 : iBGP PEERING & ROUTE PROPAGATION

In this scenario, **vEdge-BDG1** and **vEdge-BDG2** are configured within the same autonomous system (AS) 65040. **OSPF** is utilized for reachability between each LAN-facing prefix in **SITE-BANDUNG**.



To achieve iBGP peering using **Loopback IP addresses**, OSPF is used for reachability. However, since the OSPF administrative distance (AD) is smaller (110) compared to iBGP (200), OSPF routes would typically be installed in the routing table. Therefore, OSPF's AD is increased to prioritize iBGP routes.

// via CLI

1. **[vEdge-BDG1]**
2. Configure Service VPN 2, use OSPF routing for reachability of each prefix and loopback addresses for peering and propagation processes, then increase the administrative distance.

```

vpn 2
router
ospf
router-id 40.40.40.1
default-information originate
distance intra-area 210

```

<< affect the selected route to become BGP.

```
area 0
interface ge0/3
cost 5
exit
interface loopback0
exit
exit
!
bgp 65040
router-id 40.40.40.1
address-family ipv4-unicast
redistribute ospf 1                                << agar traffic yang dipelajari dari site lain di advertise ke BGP
!
neighbor 40.40.40.2
description To_vEdge-BDG2_iBGP
no shutdown
remote-as 65040
update-source loopback0
next-hop-self
!
neighbor 40.40.40.100
no shutdown
remote-as 65040
update-source loopback0
next-hop-self
!
interface ge0/3
ip address 172.16.42.1/24
no shutdown
vrrp 40
```

```

priority 200
track-omp
ipv4 172.16.42.254
!
dhcp-server
address-pool 172.16.42.0/24
exclude 172.16.42.1 172.16.42.2 172.16.42.254
options
default-gateway 172.16.42.254
dns-servers 8.8.8.8
!
interface loopback0
ip address 40.40.40.1/24
no shutdown

```

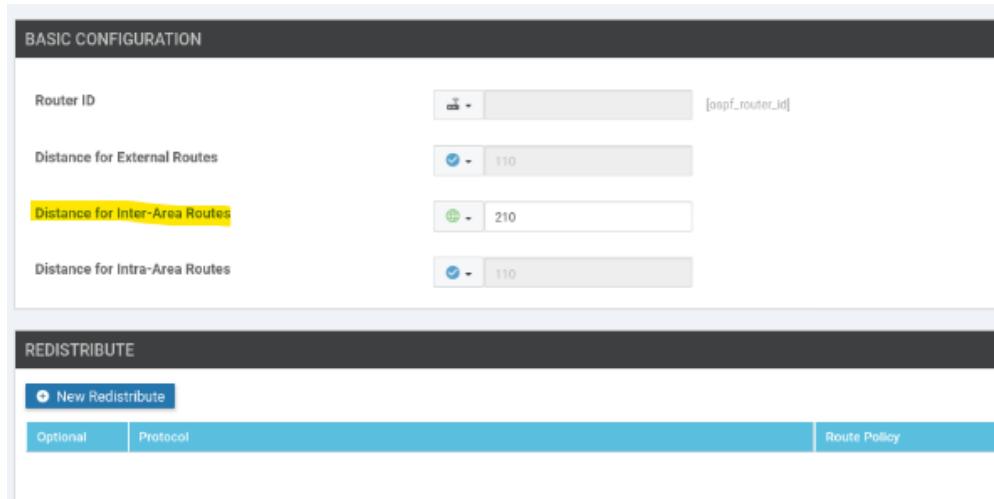
// via Templates

1. [vEdge-BDG2]

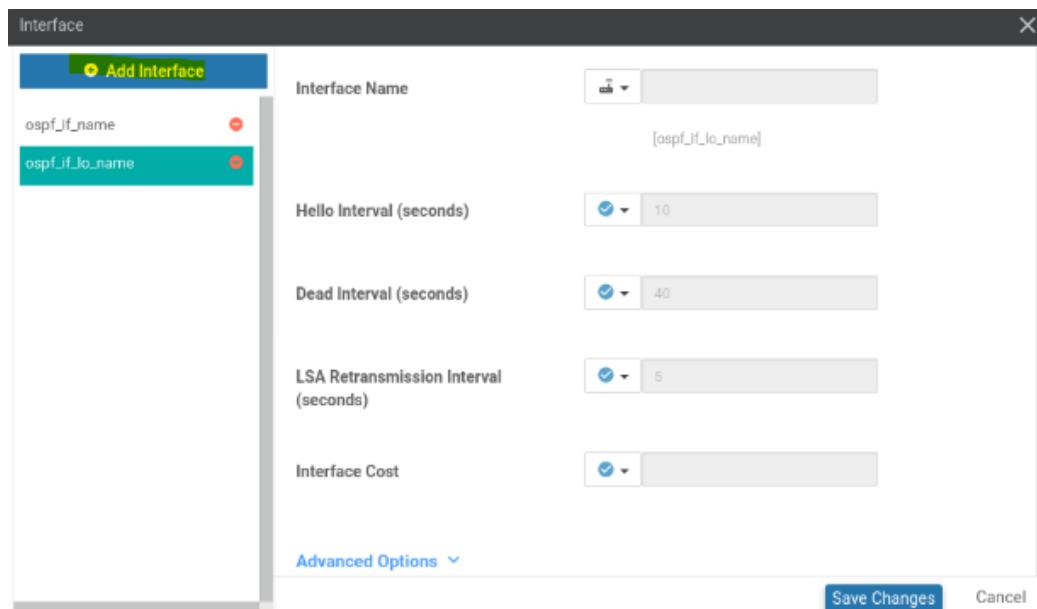
Copy the feature template of VPN 0 and VPN 1 used by the device template for Palembang Site. **Don't forget to change the VPN template to 2**, use the VRRP feature, and apply a similar DHCP IP configuration for Bandung Site. Remove any configuration specific to the site that previously used the template to ensure the device template's prior configurations remain unaffected. Add a new template for the VPN 2 loopback interface.

vEdge_BDG_VPN0_Int_Ge0/0_Template	vEdge_BDG_VPN0_Int_Ge0/0_Template	WAN Edge Interface	vEdge Cloud
vEdge_BDG_VPN0_Int_Ge0/1_Template	vEdge_BDG_VPN0_Int_Ge0/1_Template	WAN Edge Interface	vEdge Cloud
vEdge_BDG_VPN0 OSPF_Template	vEdge_BDG_VPN0 OSPF_Template	OSPF	vEdge Cloud
vEdge_BDG_VPN0_Template	vEdge_BDG_VPN0_Template	WAN Edge VPN	vEdge Cloud
vEdge_BDG_VPN2_DHCP_Templates	vEdge_Palembang_Site_DH...	DHCP Server	vEdge Cloud
vEdge_BDG_VPN2_Int_Loopback_Template	vEdge_BDG_VPN2_Int_Loo...	WAN Edge Interface	vEdge Cloud
vEdge_BDG_VPN2_Int_Template	vEdge_BDG_VPN2_Int_Tem...	WAN Edge Interface	vEdge Cloud
vEdge_BDG_VPN2 OSPF_Template	vEdge_BDG_VPN2 OSPF_T...	OSPF	vEdge Cloud
vEdge_BDG_VPN2_Template	vEdge_BDG_VPN2_Template	WAN Edge VPN	vEdge Cloud

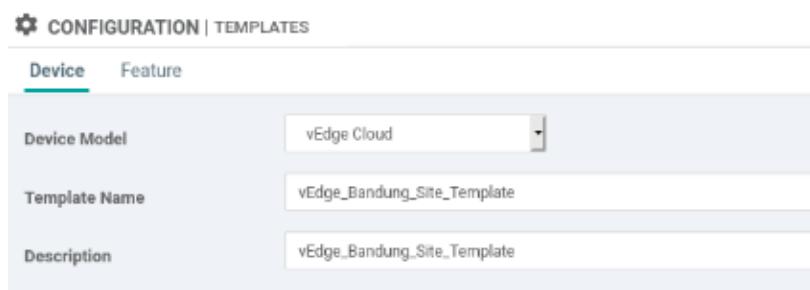
2. In the VPN 2 OSPF template, do not perform 'redistribute OMP' because the function of OSPF here is only for reachability at Bandung Site. Ensure all interconnections use BGP. Increase the administrative distance of OSPF to be greater than iBGP (200) and leave the redistribute option empty.



3. On the VPN 2 OSPF template, add the Loopback0 interface to be included in OSPF.



4. Buat device template baru untuk Bandung Site.



5. Basic Configuration

Basic Information

System *	vEdge_System_Template	
Logging*	Factory_Default_Logging_Template	
AAA *	Factory_Default_AAA_Template	BFD *
OMP *	Factory_Default_vEdge_OMP_Template	Security *

6. Transport & Management VPN

Transport & Management VPN

VPN 0 *	vEdge_BDG_VPN0_Template
OSPF	vEdge_BDG_VPN0 OSPF_Template
VPN Interface	vEdge_BDG_VPN0_Int_Ge0/0_Template
VPN Interface	vEdge_BDG_VPN0_Int_Ge0/1_Template
VPN 512 *	vEdge_VPN512_Template
VPN Interface	vEdge_VPN512_Int_eth0_Template

7. Service VPN

Service VPN + Service VPN ▾

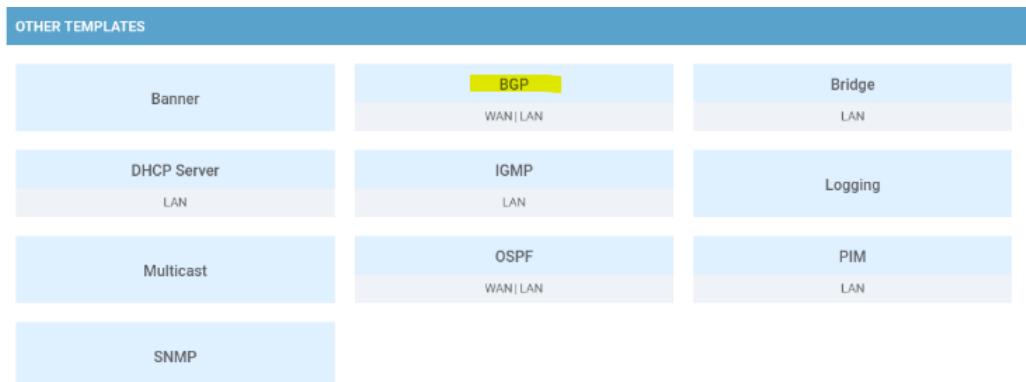
VPN	vEdge_BDG_VPN2_Template	
OSPF	vEdge_BDG_VPN2 OSPF_Template	
VPN Interface	vEdge_BDG_VPN2_Int_Template	+ Sub-Templates ▾
DHCP Server	vEdge_BDG_VPN2_DHCP_Templates	+ Sub-Templates ▾
VPN Interface	vEdge_BDG_VPN2_Int_Loopback_Template	+ Sub-Templates ▾

8. Additional Templates, then ‘Create’

Additional Templates

Banner	vEdge_Banner_Template
Policy	Choose...
SNMP	Choose...
Security Policy	Choose...

9. Create and configure the BGP feature template for VPN 2.



10. Name and describe the BGP feature template being created.

CONFIGURATION | TEMPLATES

Device	Feature
Feature Template > Add Template > BGP	
Device Type	vEdge Cloud
Template Name	vEdge_BDG_VPN2_BGP_Template
Description	vEdge_BDG_VPN2_BGP_Template

11. In the ‘Basic Configuration’ section, specify the BGP AS number for Bandung Site, using the Global option, then leave ‘Router-id’ with the device-specific option.

BASIC CONFIGURATION

Shutdown	<input checked="" type="radio"/> Yes <input type="radio"/> No
AS Number	65040
Router ID	[bgp_VPN2_router_id]
Propagate AS Path	<input checked="" type="radio"/> On <input type="radio"/> Off
Internal Routes Distance	200
Local Routes Distance	20
External Routes Distance	20

12. In the ‘Unicast Address Family’ section, select IPv4. Here, we can perform redistribution, advertise prefixes, and aggregate addresses. Redistribute OMP to allow prefixes learned at other sites to be advertised to BGP.

13. In the ‘Neighbor’ section, perform iBGP peering with vEdge-BDG1. Add a ‘New Neighbor’, then under Address, Description, and Remote-AS, choose the device-specific option, and activate the Address Family.

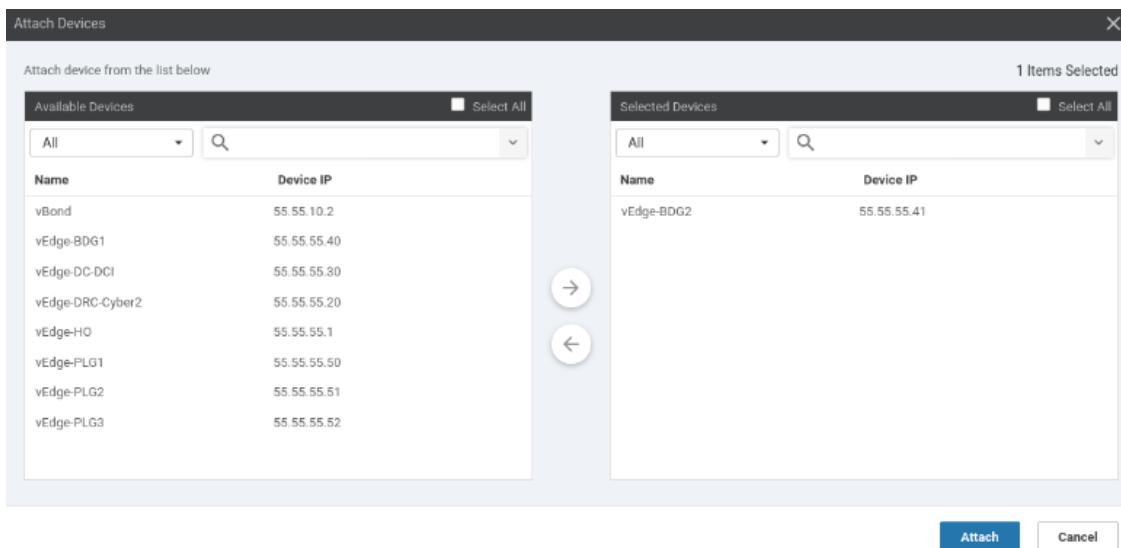
14. Still in the ‘Neighbor’ section, click ‘Advance Options’, activate Next-Hop Self, and use the device-specific option for the Source Interface Address for the peering process with the loopback address. Finally, ‘Add’ and ‘Save’ the feature template

15. Add another ‘New Neighbor’ for BGP peering with SITE-BANDUNG2, following the same steps as above.

NEIGHBOR			
New Neighbor			
Optional	Address	Description	Remote AS
<input checked="" type="checkbox"/>	[bgp_neighbor_address]	[bgp_neighbor_description]	[bgp_neighbor_remote_as]
<input checked="" type="checkbox"/>	[bgp_neighbor_lan_address]	[bgp_neighbor_lan_description]	[bgp_neighbor_lan_remote_as]

16. Add the created BGP feature template to the device template from Bandung Site, specifically under Service VPN 2, then ‘Update’.

17. Add vEdge-BDG2 to the Bandung Site device template that was created.



18. Edit all parameter values related to the device template according to the topology and scenario for vEdge-BDG2.

Address Pool(dhcp_address_pool)	172.16.42.0/24
Exclude Addresses(dhcp_address_exclude)	172.16.42.1,172.16.42.2,172.16.42.254
Default Gateway(dhcp_default_gateway)	172.16.42.254
DNS Servers(dhcp_dns_server)	8.8.8.8
Router ID(ospf_router_id)	40.40.40.2
Area Number(ospf_area_a_num)	0
Interface Name(ospf_if_name)	ge0/2
Interface Name(ospf_if_lo_name)	loopback0
Interface Cost(ospf_cost)	10
Router ID(bgp_VPN2_router_id)	40.40.40.2
Address(bgp_neighbor_address)	40.40.40.1
Description(bgp_neighbor_description)	To_vEdge-BDG1_IBGP_Peering
Remote AS(bgp_neighbor_remote_as)	65040
Source Interface Name(bgp_neighbor_if_name)	loopback0
Address(bgp_neighbor_lan_address)	40.40.40.100
Description(bgp_neighbor_lan_description)	To LAN Facing_Site Bandung
Remote AS(bgp_neighbor_lan_remote_as)	65040
Source Interface Name(bgp_neighbor_if_lo_name)	loopback0
Protocol(bgp_redistribute_protocol)	ospf

19. Verify Config Preview before pushing; if it's safe, click 'Configure Device'.

```

ospf
router-id 40.40.40.2
default-information originate
timers spf 200 1000 10000
area 0
  interface ge0/2
    cost 10
  exit
  interface loopback0
  exit
exit
!
bgp 65040
router-id 40.40.40.2
neighbor 40.40.40.1
  description To_vEdge-BDG1_iBGP_Peering
  no shutdown
  remote-as 65040
  update-source loopback0
  next-hop-self
  address-family ipv4-unicast
!
neighbor 40.40.40.100
  description "To LAN Facing_Site Bandung"
  no shutdown
  remote-as 65040
  update-source loopback0
  next-hop-self
  address-family ipv4-unicast
!
```

Back**Configure Devices****Cancel**

20. Thus, the configuration is successfully pushed to vEdge-PLG2.

```

vEdge-BDG2# show run vpn 2
vpn 2
  router
    ospf
      router-id 40.40.40.2
      default-information originate
      timers spf 200 1000 10000
      area 0
        interface ge0/2
          cost 10
        exit
        interface loopback0
        exit
      exit
    !
    bgp 65040
      router-id 40.40.40.2
      address-family ipv4-unicast
        redistribute ospf
      !
      neighbor 40.40.40.1
        description To_vEdge-BDG1_iBGP_Peering
        no shutdown
        remote-as 65040
        update-source loopback0
        next-hop-self
        address-family ipv4-unicast
      !
```

VPN 2 : iBGP Verification

For verification purposes, set up DRC-Cyber2 site so that BGP can advertise prefixes from OMP routes (prefixes from DRC-Cyber2) using redistribute OMP

// show ip route bgp (vEdge-BDG)

```
vEdge-BDG1# show ip route bgp
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

      VPN      PREFIX          PROTOCOL      PROTOCOL      NEXTHOP      NEXTHOP
      SUB TYPE      IF NAME      ADDR
-----+-----+-----+-----+-----+-----+
  2    192.168.46.0/24    bgp           i            -        40.40.40.100
  2    192.168.47.0/24    bgp           i            -        40.40.40.100
  2    192.168.48.0/24    bgp           i            -        40.40.40.100
```

```
vEdge-BDG2# show ip route bgp
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

      VPN      PREFIX          PROTOCOL      PROTOCOL      NEXTHOP      NEXTHOP
      SUB TYPE      IF NAME      ADDR
-----+-----+-----+-----+-----+
  2    192.168.46.0/24    bgp           i            -        40.40.40.100
  2    192.168.47.0/24    bgp           i            -        40.40.40.100
  2    192.168.48.0/24    bgp           i            -        40.40.40.100
```

Has routes advertised on SITE-BANDUNG2

// show ip route (SITE-BANDUNG2)

```
SITE-BANDUNG2#show ip route bgp
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
      a - application route
      + - replicated route, % - next hop override, p - overrides from Pfr
      Gateway of last resort is 172.16.42.254 to network 0.0.0.0

          20.0.0.0/32 is subnetted, 1 subnets
          B    20.20.20.100 [200/1000] via 40.40.40.1, 00:28:42
              172.16.0.0/16 is variably subnetted, 3 subnets, 2 masks
              B    172.16.22.0/24 [200/1000] via 40.40.40.1, 00:29:37
              B    192.168.26.0/24 [200/1000] via 40.40.40.1, 00:28:42
              B    192.168.27.0/24 [200/1000] via 40.40.40.1, 00:28:42
              B    192.168.28.0/24 [200/1000] via 40.40.40.1, 00:28:42
```

Redistributed omp routes are accepted as BGP

```
// show bgp summary
```

```
vEdge-BDG1# show bgp summary
vpn          2
bgp-router-identifier 40.40.40.1
local-as      65040
rib-entries   15
rib-memory    1680
total-peers   2
peer-memory   9632
Local-soo     So0:0:40
ignore-soo

NEIGHBOR     AS   MSG RCVD   MSG SENT   OUT Q   UPTIME           PREFIX RCVD   PREFIX VALID   PREFIX INSTALLED STATE
-----
```

NEIGHBOR	AS	MSG RCVD	MSG SENT	OUT Q	UPTIME	PREFIX RCVD	PREFIX VALID	PREFIX INSTALLED	STATE
40.40.40.2	65040	61	66	0	0:00:59:04	0	0	0	established
40.40.40.100	65040	68	65	0	0:00:58:58	3	3	3	established

```
vEdge-BDG2# show bgp sum
vpn          2
bgp-router-identifier 40.40.40.2
local-as      65040
rib-entries   15
rib-memory    1680
total-peers   2
peer-memory   9632
Local-soo     So0:0:40
ignore-soo

NEIGHBOR     AS   MSG RCVD   MSG SENT   OUT Q   UPTIME           PREFIX RCVD   PREFIX VALID   PREFIX INSTALLED STATE
-----
```

NEIGHBOR	AS	MSG RCVD	MSG SENT	OUT Q	UPTIME	PREFIX RCVD	PREFIX VALID	PREFIX INSTALLED	STATE
40.40.40.1	65040	65	63	0	0:00:59:40	0	0	0	established
40.40.40.100	65040	69	63	0	0:00:59:39	3	3	3	established

```
SITE-BANDUNG2#show bgp sum
BGP router identifier 40.40.40.100, local AS number 65040
BGP table version is 11, main routing table version 11
8 network entries using 1152 bytes of memory
13 path entries using 1092 bytes of memory
2/2 BGP path/bestpath attribute entries using 320 bytes of memory
1 BGP extended community entries using 24 bytes of memory
0 BGP route-map cache entries using 0 bytes of memory
0 BGP filter-list cache entries using 0 bytes of memory
BGP using 2588 total bytes of memory
BGP activity 9/1 prefixes, 14/1 paths, scan interval 60 secs

Neighbor      V      AS MsgRcvd MsgSent   TblVer InQ OutQ Up/Down State/PfxRcd
40.40.40.1    4      65040  67      71        11     0      0 01:00:13      5
40.40.40.2    4      65040  64      70        11     0      0 01:00:18      5
```

There is 5 routes from SITE-BANDUNG2 perspective

```
// show received-routes (SITE-BANDUNG2)
```

```
SITE-BANDUNG2#show bgp ipv4 unicast neighbors 40.40.40.1 received-routes
BGP table version is 11, local router ID is 40.40.40.100
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
              t secondary path,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

      Network          Next Hop          Metric LocPrf Weight Path
*>i  20.20.20.100/32  40.40.40.1          1000    50      0 ?
*>i  172.16.22.0/24  40.40.40.1          1000    50      0 ?
*>i  192.168.26.0    40.40.40.1          1000    50      0 ?
*>i  192.168.27.0    40.40.40.1          1000    50      0 ?
*>i  192.168.28.0    40.40.40.1          1000    50      0 ?
```

All IPs on the DRC2-Cryber2 site are advertised because they are known as omp

```
// show ip bgp (SITE-BANDUNG2)
```

```
SITE-BANDUNG2#show ip bgp
BGP table version is 11, local router ID is 40.40.40.100
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
               r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
               x best-external, a additional-path, c RIB-compressed,
               t secondary path,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

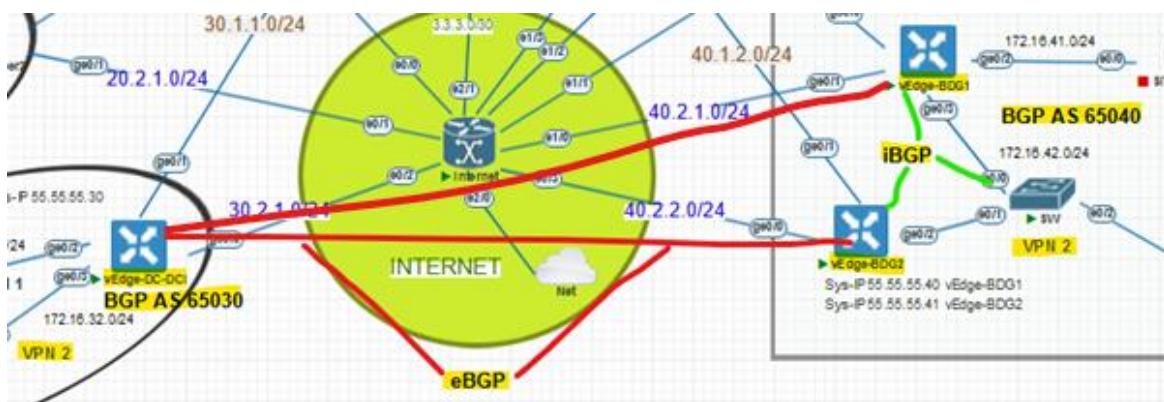
      Network          Next Hop        Metric LocPrf Weight Path
* i  20.20.20.100/32  40.40.40.2      1000   50      0  ?
* >i                         40.40.40.1      1000   50      0  ?
* i  172.16.22.0/24   40.40.40.2      1000   50      0  ?
* >i                         40.40.40.1      1000   50      0  ?
* i  192.168.26.0    40.40.40.2      1000   50      0  ?
* >i                         40.40.40.1      1000   50      0  ?
* i  192.168.27.0    40.40.40.2      1000   50      0  ?
* >i                         40.40.40.1      1000   50      0  ?
* i  192.168.28.0    40.40.40.2      1000   50      0  ?
* >i                         40.40.40.1      1000   50      0  ?
*>  192.168.46.0     0.0.0.0          0        32768 i
*>  192.168.47.0     0.0.0.0          0        32768 i
*>  192.168.48.0     0.0.0.0          0        32768 i
```

```
// ping
```

```
SITE-BANDUNG2#ping 192.168.26.10
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.26.10, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 41/56/68 ms
SITE-BANDUNG2#ping 192.168.27.10
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.27.10, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 43/59/71 ms
SITE-BANDUNG2#ping 192.168.28.10
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.28.10, timeout is 2 seconds:
!!!!!
```

VPN 2 : eBGP PEERING & ROUTE PROPAGATION

In this eBGP scenario, we will establish peering and route propagation between autonomous systems (AS) from Bandung Site and DC-DCI Site. The devices vEdge-BDG2 and vEdge-DC-DCI are using the Template method, but for vEdge-BDG1, we will continue using the CLI. We only need to add a few parameters for vEdge-BDG1 and vEdge-BDG2 related to the neighbor peering to vEdge-DC-DCI, and then add the BGP setup from scratch for vEdge-DC-DCI.



If previously we used the prefix from DRC-Cyber2 as an example to obtain prefixes in the iBGP scenario, here **we will focus on prefixes that all originate from BGP (not from redistribute OMP)**.

// vEdge-BDG1

Add a neighbor to the vEdge-DC-DCI device, because we know that eBGP has a TTL of 1. Configure `ebgp-mulhop` to increase the TTL for the eBGP peering process (since loopback is used, there will be 2 hops/TTL). Then, remove the configuration of ‘redistribute omp’.

```
router bgp 65040
  address-family ipv4-unicast
  no redistribute omp
  !
  neighbor 30.30.30.1
  description To_vEdge-DC-DCI_eBGP
```

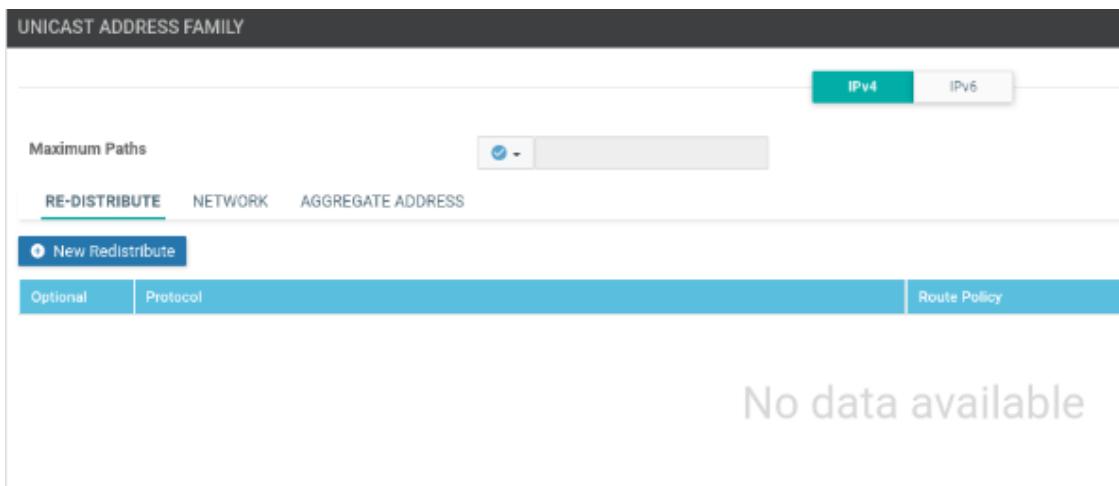
```

no shutdown
remote-as 65030
update-source loopback0
ebgp-multipath 2
!

```

// vEdge-BDG2

1. In the BGP VPN2 feature template for Bandung Site, specifically in the ‘Unicast Address Family’ section, remove redistribute OMP.



2. Next, we only need to add a neighbor to the loopback of vEdge-BDG2. Go to the BGP VPN 2 feature template for Bandung Site and select ‘New Neighbor’.



3. Set Address, Description, and Remote-AS to device-specific options, then activate Address-family.

Address	<input type="text"/> [bgp_neighbor_DC_address]
Description	<input type="text"/> [bgp_neighbor_description]
Remote AS	<input type="text"/> [bgp_neighbor_remote_as]
Address Family	<input checked="" type="radio"/> On <input type="radio"/> Off
Address Family	IPv4 unicast
Maximum Number of Prefixes	<input type="text"/>
Route Policy In	<input checked="" type="radio"/> On <input type="radio"/> Off
Route Policy Out	<input checked="" type="radio"/> On <input type="radio"/> Off
Shutdown	<input checked="" type="radio"/> Yes <input type="radio"/> No

4. Still in the ‘Neighbor’ section, expand ‘Advanced Options’, then change the Source Interface to ‘Name’ to update the source interface. Since this is eBGP, assign a value to EBGP Multihop to increase the TTL for the peering process. Once done, click ‘Add’ and then ‘Update’.

Next-Hop Self	<input checked="" type="radio"/> On <input type="radio"/> Off
Send Community	<input checked="" type="radio"/> On <input type="radio"/> Off
Send Extended Community	<input checked="" type="radio"/> On <input type="radio"/> Off
Negotiate Capability	<input checked="" type="radio"/> On <input type="radio"/> Off
Source Interface	<input type="radio"/> Address <input checked="" type="radio"/> Name
Source Interface Name	<input type="text"/> [bgp_neighbor_lo-dc_name]
EBGP Multihop	<input type="text"/> 2
Password	<input type="text"/>

5. vEdge-BDG2 will now have 3 neighbors; proceed to ‘Update’.

NEIGHBOR		
	Address	Description
Optional	[bgp_neighbor_address]	[bgp_neighbor_description]
Optional	[bgp_neighbor_lan_address]	[bgp_neighbor_lan_description]
Optional	[bgp_neighbor_DC_address]	[bgp_neighbor_dc_description]

6. Edit the Device Template to add the updated parameter values for the vEdge-BDG2 device. Once complete, click ‘Update’ and then ‘Next’.

Address(bgp_neighbor_DC_address)	30.30.30.1
Description(bgp_neighbor_dc_description)	To_vEdge-DC-DCI_eBGP
Remote AS(bgp_neighbor_dc_remote_as)	65030
Source Interface Name(bgp_neighbor_lo-dc_name)	loopback0

Generate Password **Update**

7. Verify the configuration before pushing it to vEdge-BDG2. If everything is safe, proceed to ‘Configure Devices’.
8. The configuration has been successfully pushed to vEdge-BDG2.

```
neighbor 30.30.30.1
description To_vEdge-DC-DCI_eBGP
no shutdown
remote-as 65030
update-source loopback0
ebgp-multipath 2
address-family ipv4-unicast
!
```

// vEdge-DC-DCI

1. Copy all related feature templates for VPN 2 from the previous vEdge-BDG2. Remove any configurations specific to vEdge-BDG2. On vEdge-DC-DCI, there is only one path from the LAN perspective, so don’t forget to remove the VRRP configuration.

vEdge_DC_VPN2_BGP_Template	vEdge_DC_VPN2_BGP_Te...	BGP
vEdge_DC_VPN2_DHCP_Templates	vEdge_DC_VPN2_DHCP_Te...	DHCP Server
vEdge_DC_VPN2_Int_Loopback_Template	vEdge_BDG_VPN2_Int_Loo...	WAN Edge Interface
vEdge_DC_VPN2_Int_Template	vEdge_DC_VPN2_Int_Temp...	WAN Edge Interface
vEdge_DC_VPN2 OSPF_Template	vEdge_DC_VPN2 OSPF_Te...	OSPF
vEdge_DC_VPN2_Template	vEdge_DC_VPN2_Template	WAN Edge VPN

2. Focus on the BGP feature template. There are 3 neighbors on this site, same as before. However, the neighbor leading to Bandung Site does not use Next-hop self. We need to add EBGP Multihop for the neighbor leading to that site. Once done, click ‘Update’.

The screenshot shows a 'NEIGHBOR' configuration page. At the top, there are tabs for 'IPv4' and 'IPv6'. Below the tabs, a 'New Neighbor' button is highlighted. A table lists three neighbors:

Optional	Address	Description
<input checked="" type="checkbox"/>	[bgp_neighbor1_address]	[bgp_neighbor1_description]
<input checked="" type="checkbox"/>	[bgp_neighbor1_an_address]	[bgp_neighbor1_an_description]
<input checked="" type="checkbox"/>	[bgp_neighbor2_address]	[bgp_neighbor2_description]

At the bottom right of the form are 'Update' and 'Cancel' buttons.

3. Since the device template for DC-DCI already exists, we only need to include all relevant feature templates for VPN 2, including OSPF and BGP. Once complete, click ‘Update’.

The screenshot shows the 'Service VPN' tab of a device template configuration page. The tab is selected and highlighted in green. The page includes sections for Basic Information, Transport & Management VPN, Service VPN, and Additional Templates. Under the Service VPN tab, the following feature templates are selected:

- VPN: vEdge_DC_VPN2_Template
- BGP: vEdge_DC_VPN2_BGP_Template
- OSPF: vEdge_DC_VPN2_OSPF_Template
- VPN Interface: vEdge_DC_VPN2_Int_Template
- DHCP Server: vEdge_DC_VPN2_DHCP_Templates
- VPN Interface: vEdge_DC_VPN2_Int_Loopback_Template

For each selected template, there is a 'Sub-Templates' dropdown menu.

4. Fill in all required parameter values for the newly added feature template for VPN 2.

Update Device Template

Variable List (Hover over each field for more information)	
Router ID(ospf_router_id)	30.30.30.1
Area Number(ospf_area_a_num)	0
Interface Name(ospf_if_name)	ge0/3
Interface Name(ospf_if_lo_name)	loopback0
Interface Cost(ospf_cost)	10
AS Number(bgp_as_num)	65030
Router ID(bgp_VPN2_router_id)	30.30.30.1
Address(bgp_neighbor1_address)	40.40.40.1
Address(bgp_neighbor1_lan_address)	30.30.30.100
Address(bgp_neighbor2_address)	40.40.40.2
Description(bgp_neighbor1_description)	To BDG1
Description(bgp_neighbor1_lan_description)	To LAN Facing_DC
Description(bgp_neighbor2_description)	To BDG2
Remote AS(bgp_neighbor1_remote_as)	65040
Remote AS(bgp_neighbor1_lan_remote_as)	65030
Remote AS(bgp_neighbor2_remote_as)	65040
Source Interface Name(bgp_neighbor1_if_name)	loopback0

Buttons: Generate Password, Update, Cancel

5. Verify the configuration before pushing it to the vEdge-DC-DCI device. If everything is safe, click ‘Configure Devices’.

```

bgp 65030
  router-id 30.30.30.1
  neighbor 30.30.30.100
    description "To LAN Facing_DC"
    no shutdown
    remote-as 65030
    update-source loopback0
    next-hop-self
    address-family ipv4-unicast
  !
  !
  neighbor 40.40.40.1
    description "To BDG1"
    no shutdown
    remote-as 65040
    update-source loopback0
    no next-hop-self
    ebgp-multihop 2
    address-family ipv4-unicast
  !
  !
  neighbor 40.40.40.2
    description "To BDG2"
    no shutdown
    remote-as 65040
    update-source loopback0
    ebgp-multihop 2
  !

```

Buttons: Back, Configure Devices, Cancel

6. Thus, the template configuration has been successfully pushed to the device.

```
vEdge-DC-DCI# show run vpn 2
vpn 2
  router
    ospf
      router-id 30.30.30.1
      default-information originate
      timers spf 200 1000 10000
      area 0
        interface ge0/3
        cost 10
      exit
      interface loopback0
      exit
    exit
!
bgp 65030
  router-id 30.30.30.1
  neighbor 30.30.30.100
    description "To LAN Facing_DC"
  no shutdown
  remote-as 65030
  update-source loopback0
  next-hop-self
  address-family ipv4-unicast
!
```

VPN 2 : eBGP Verification

// show ip route bgp (vEdge-BDG & vEdge-DC-DCI)

```
vEdge-BDG1# show ip route bgp
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
  --  --  --  --  --
  2   192.168.36.0/24  bgp          i         -        30.30.30.1
  2   192.168.37.0/24  bgp          i         -        30.30.30.1
  2   192.168.38.0/24  bgp          i         -        30.30.30.1
  2   192.168.46.0/24  bgp          i         -        40.40.40.100
  2   192.168.47.0/24  bgp          i         -        40.40.40.100
  2   192.168.48.0/24  bgp          i         -        40.40.40.100
```

```
vEdge-BDG2# show ip route bgp
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
-----
2    192.168.36.0/24  bgp         i          -          30.30.30.1
2    192.168.37.0/24  bgp         i          -          30.30.30.1
2    192.168.38.0/24  bgp         i          -          30.30.30.1
2    192.168.46.0/24  bgp         i          -          40.40.40.100
2    192.168.47.0/24  bgp         i          -          40.40.40.100
2    192.168.48.0/24  bgp         i          -          40.40.40.100
```

```
vEdge-DC-DCI# show ip route bgp
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
-----
2    192.168.36.0/24  bgp         i          -          30.30.30.100
2    192.168.37.0/24  bgp         i          -          30.30.30.100
2    192.168.38.0/24  bgp         i          -          30.30.30.100
2    192.168.46.0/24  bgp         i          -          40.40.40.2
2    192.168.47.0/24  bgp         i          -          40.40.40.2
2    192.168.48.0/24  bgp         i          -          40.40.40.2
```

Have mutual routes advertised from the Lan Facing perspective

// show ip route (SITE-BANDUNG2 & DC2-DCI)

```
DC2-DCI#show ip route bgp
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
      a - application route
      + - replicated route, % - next hop override, p - overrides from Pfr

Gateway of last resort is 172.16.32.1 to network 0.0.0.0

B    192.168.46.0/24 [200/0] via 30.30.30.1, 00:57:15
B    192.168.47.0/24 [200/0] via 30.30.30.1, 00:57:15
B    192.168.48.0/24 [200/0] via 30.30.30.1, 00:57:15
```

```

SITE-BANDUNG2#show ip route bgp
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
      a - application route
      + - replicated route, % - next hop override, p - overrides from PfR

Gateway of last resort is 172.16.42.254 to network 0.0.0.0

B    192.168.36.0/24 [200/0] via 40.40.40.1, 00:57:35
B    192.168.37.0/24 [200/0] via 40.40.40.1, 00:57:35
B    192.168.38.0/24 [200/0] via 40.40.40.1, 00:57:35

```

Prefixes from each other are installed into the routing table as BGP

// show bgp summary

```

vEdge-BDG1# show bgp summary
vpn          2
bgp-router-identifier 40.40.40.1
local-as      65040
rib-entries   11
rib-memory    1232
total-peers   3
peer-memory   14448
Local-soo     So0:0:40
ignore-soo

NEIGHBOR     AS      MSG      MSG      OUT      PREFIX      PREFIX      PREFIX
              RCV'D    SENT     Q        UPTIME    RCV'D    VALID    INSTALLED STATE
-----+-----+-----+-----+-----+-----+-----+-----+-----+
30.30.30.1   65030  123      121      0        0:01:53:11  3        3        3        established
40.40.40.2   65040  399      400      0        0:06:30:24  3        3        0        established
40.40.40.100 65040  429      402      0        0:01:28:11  3        3        3        established

```

```

vEdge-BDG2# show bgp summary
vpn          2
bgp-router-identifier 40.40.40.2
local-as      65040
rib-entries   11
rib-memory    1232
total-peers   3
peer-memory   14448
Local-soo     So0:0:40
ignore-soo

NEIGHBOR     AS      MSG      MSG      OUT      PREFIX      PREFIX      PREFIX
              RCV'D    SENT     Q        UPTIME    RCV'D    VALID    INSTALLED STATE
-----+-----+-----+-----+-----+-----+-----+-----+-----+
30.30.30.1   65030  120      122      0        0:01:54:05  3        3        3        established
40.40.40.1   65040  399      401      0        0:06:31:04  3        3        0        established
40.40.40.100 65040  430      402      0        0:01:28:50  3        3        3        established

```

```

vEdge-DC-DCI# show bgp sum
vpn          2
bgp-router-identifier 30.30.30.1
local-as      65030
rib-entries   11
rib-memory    1232
total-peers   3
peer-memory   14448
Local-soo     So0:0:30
ignore-soo

NEIGHBOR     AS      MSG      MSG      OUT      PREFIX      PREFIX      PREFIX
              RCV'D    SENT     Q        UPTIME    RCV'D    VALID    INSTALLED STATE
-----+-----+-----+-----+-----+-----+-----+-----+-----+
30.30.30.100 65030  110      103      0        0:01:31:52  3        3        3        established
40.40.40.1   65040  120      125      0        0:01:54:25  3        3        0        established
40.40.40.2   65040  121      122      0        0:01:54:40  3        3        3        established

```

```
SITE-BANDUNG2#show ip bgp sum
BGP router identifier 40.40.40.100, local AS number 65040
BGP table version is 16, main routing table version 16
6 network entries using 864 bytes of memory
9 path entries using 756 bytes of memory
2/2 BGP path/bestpath attribute entries using 320 bytes of memory
1 BGP AS-PATH entries using 24 bytes of memory
0 BGP route-map cache entries using 0 bytes of memory
0 BGP filter-list cache entries using 0 bytes of memory
BGP using 1964 total bytes of memory
BGP activity 6/0 prefixes, 15/6 paths, scan interval 60 secs

Neighbor      V          AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down State/PfxRcd
40.40.40.1    4        65040     95     103       16     0     0 01:29:59      3
40.40.40.2    4        65040     95     103       16     0     0 01:29:58      3
```

```
DC2-DCI#show ip bgp sum
BGP router identifier 30.30.30.100, local AS number 65030
BGP table version is 16, main routing table version 16
6 network entries using 864 bytes of memory
6 path entries using 504 bytes of memory
2/2 BGP path/bestpath attribute entries using 328 bytes of memory
1 BGP AS-PATH entries using 24 bytes of memory
0 BGP route-map cache entries using 0 bytes of memory
0 BGP filter-list cache entries using 0 bytes of memory
BGP using 1720 total bytes of memory
BGP activity 6/0 prefixes, 6/0 paths, scan interval 60 secs

Neighbor      V          AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down State/PfxRcd
30.30.30.1    4        65030     96     106       16     0     0 01:32:58      3
```

// show received-routes (SITE-BANDUNG2 & DC2-DCI)

```
SITE-BANDUNG2#show bgp ipv4 unicast neighbors 40.40.40.1 received-routes
BGP table version is 16, local router ID is 40.40.40.100
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
              t secondary path,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

      Network          Next Hop            Metric LocPrf Weight Path
*>i 192.168.36.0    40.40.40.1           100      0 65030 i
*>i 192.168.37.0    40.40.40.1           100      0 65030 i
*>i 192.168.38.0    40.40.40.1           100      0 65030 i

Total number of prefixes 3
```

```
DC2-DCI#show bgp ipv4 unicast neighbors 30.30.30.1 received-routes
BGP table version is 16, local router ID is 30.30.30.100
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
              t secondary path,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

      Network          Next Hop            Metric LocPrf Weight Path
*>i 192.168.46.0    30.30.30.1           100      0 65040 i
*>i 192.168.47.0    30.30.30.1           100      0 65040 i
*>i 192.168.48.0    30.30.30.1           100      0 65040 i

Total number of prefixes 3
```

// show ip bgp (SITE-BANDUNG2 & DC2-DCI)

```
DC2-DCI#show ip bgp
BGP table version is 16, local router ID is 30.30.30.100
Status codes: s suppressed, d damped, h history, * valid, > best, i - in
               r RIB-failure, S Stale, m multipath, b backup-path, f RT-F
               x best-external, a additional-path, c RIB-compressed,
               t secondary path,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

      Network          Next Hop           Metric LocPrf Weight Path
*-> 192.168.36.0    0.0.0.0            0        32768 i
*-> 192.168.37.0    0.0.0.0            0        32768 i
*-> 192.168.38.0    0.0.0.0            0        32768 i
*>i 192.168.46.0    30.30.30.1         100      0 65040 i
*>i 192.168.47.0    30.30.30.1         100      0 65040 i
*>i 192.168.48.0    30.30.30.1         100      0 65040 i
```

```
SITE-BANDUNG2#show ip bgp
BGP table version is 16, local router ID is 40.40.40.100
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
               r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
               x best-external, a additional-path, c RIB-compressed,
               t secondary path,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

      Network          Next Hop           Metric LocPrf Weight Path
* i 192.168.36.0    40.40.40.2         100      0 65030 i
*>i 192.168.37.0    40.40.40.1         100      0 65030 i
* i 192.168.38.0    40.40.40.2         100      0 65030 i
*>i 192.168.46.0    0.0.0.0            0        32768 i
*>i 192.168.47.0    0.0.0.0            0        32768 i
*>i 192.168.48.0    0.0.0.0            0        32768 i
```

// ping

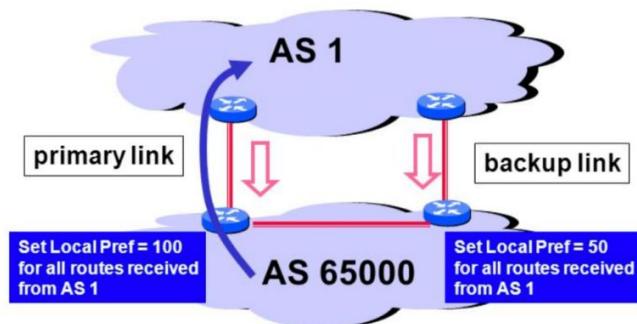
```
DC2-DCI#ping 192.168.46.10
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.46.10, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 56/66/82 ms
DC2-DCI#ping 192.168.47.10
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.47.10, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 41/53/62 ms
DC2-DCI#ping 192.168.48.10
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.48.10, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 58/66/72 ms
```

```
SITE-BANDUNG2#ping 192.168.36.10
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.36.10, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 50/65/72 ms
SITE-BANDUNG2#ping 192.168.37.10
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.37.10, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 51/66/72 ms
SITE-BANDUNG2#ping 192.168.38.10
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.38.10, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 40/51/62 ms
```

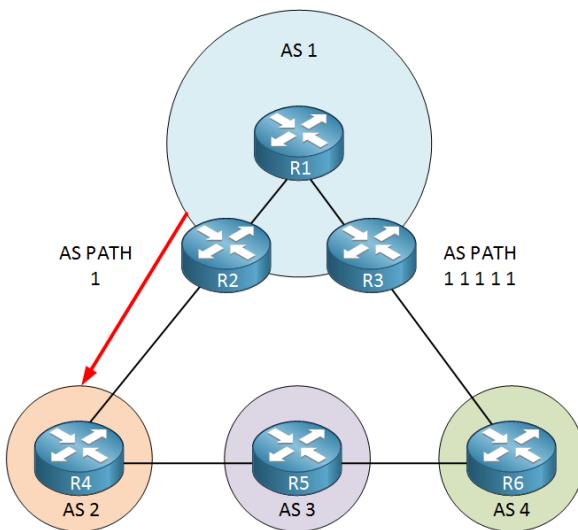
VPN 2 : BGP Localized Route Policy

BGP is a protocol that uses path attributes that can be applied to each prefix connected in BGP. Path attributes can be used as a method to influence the route taken by a prefix to its destination. In this scenario, we will apply route policies using the following path attributes:

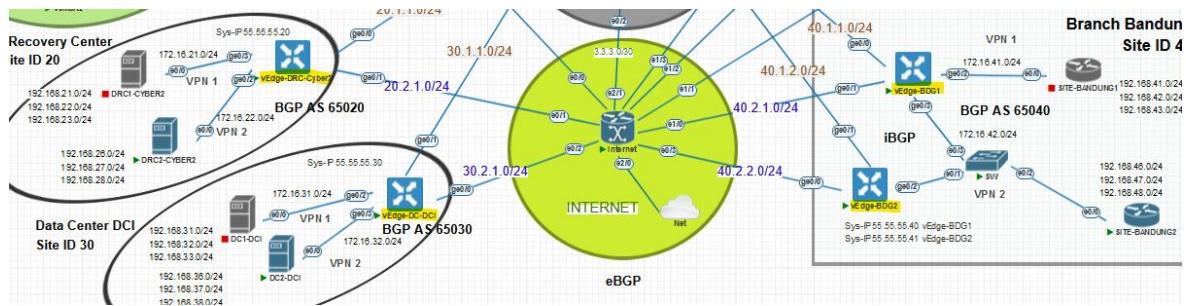
- a. **Local Preference (Outbound)**, From the perspective of our intranet, if a prefix enters the intranet, it will be set with a local preference by the BGP router. When the intranet wants to reach that prefix, in cases where there are two path options or BGP routers, **the intranet will choose the path with the highest local preference as the best route.**



- b. **AS Path (Inbound)**, for this attribute, the opposite is true. It is viewed from the perspective of the extranet. Our BGP routers (intranet) influence the paths they take by providing information regarding the AS-PATHs they should follow to reach the destination address within the intranet. **The more AS-PATHs that need to be traversed, the less likely the path will be chosen.**



// SKENARIO



In this scenario, **all configurations are performed only on vEdge-BDG1 & vEdge-BDG2**. SITE-BANDUNG already has prefixes from the vEdge-DC-DCI devices. In this scenario, we **also need prefixes from vEdge-DRC-Cyber2**. Therefore, on the vEdge-BDG devices, establish neighbor peering with vEdge-DRC-Cyber2 and advertise the prefixes from DRC Cyber2. The following are the prefixes owned by SITE-BANDUNG2 after prefixes from the DRC Cyber2 site are redistributed into BGP for the route policy scenario we are implementing:

```
SITE-BANDUNG2#show ip bgp
BGP table version is 61, local router ID is 40.40.40.100
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
               r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
               x best-external, a additional-path, c RIB-compressed,
               t secondary path,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

      Network          Next Hop            Metric LocPrf Weight Path
* i 192.168.26.0    40.40.40.2        11    100    0 65020 i
*>i                               40.40.40.1        11    100    0 65020 i
* i 192.168.27.0    40.40.40.2        11    100    0 65020 i
*>i                               40.40.40.1        11    100    0 65020 i
* i 192.168.28.0    40.40.40.2        11    100    0 65020 i
*>i                               40.40.40.1        11    100    0 65020 i
*>i 192.168.36.0    40.40.40.1        100   0       0 65030 i
* i 192.168.37.0    40.40.40.1        100   0       0 65030 i
* i 192.168.38.0    40.40.40.2        100   0       0 65030 i
*>i 192.168.38.0    40.40.40.1        100   0       0 65030 i
* i 192.168.38.0    40.40.40.2        100   0       0 65030 i
```

The points to be implemented and the configurations related to this BGP route policy include:

-- vEdge-BDG1 via CLI --

- [Local Pref] Prefix **outbond to** DRC Cyber2 Site (**192.168.2x.0**) will go through via **40.40.40.1**, with **local preference greater than default**
- [AS Path] Prefix **inbound to** Bandung Site from DRC Cyber2 Site, it is made **symmetric via 40.40.40.1 (vEdge-BDG1)** by **manipulating the AS-path prepend** sent by vEdge-BDG2 (so that the prefixes from DRC Cyber2 prefer vEdge-BDG1).

-- vEdge-BDG2 via Template --

- **[Local Pref]** Prefix **outbond to** DC-DCI Site (**192.168.3x.0**) will go through via **40.40.40.2**, with **local preference greater than default**
- **[AS Path]** Prefix **inbound to** Bandung Site from DC-DCI Site, it is made **symmetric via 40.40.40.2 (vEdge-BDG2) by manipulating the AS-path prepend** sent by vEdge-BDG1 (so that the prefixes from DC-DCI prefer vEdge-BDG2).

// vEdge-BDG1 (via CLI)

```
policy
lists
prefix-list DC-PREFIX
ip-prefix 192.168.36.0/24
ip-prefix 192.168.37.0/24
ip-prefix 192.168.38.0/24
!
prefix-list DRC-PREFIX
ip-prefix 192.168.26.0/24
ip-prefix 192.168.27.0/24
ip-prefix 192.168.28.0/24
!
prefix-list LAN-FACING-PREFIX
ip-prefix 192.168.46.0/24
ip-prefix 192.168.47.0/24
ip-prefix 192.168.48.0/24
!
!
route-policy SET_LOCALPREF_RECEIVED_PREFIX
sequence 5
match
address DRC-PREFIX
```

```
!
action accept
set
local-preference 1000          << All DRC prefixes entering will be set with a local preference of
!
1000, for outbound traffic from the intranet/site Bandung.
sequence 10
match
address DC-PREFIX
!
action accept                  << accept prefix DC too, without set local pref
!
!
default-action reject
!
route-policy DONT_PASSING_USE_OTHER_PATH
sequence 5
match
address LAN-FACING-PREFIX
!
action accept
set
as-path prepend 11111          << Adding an AS-path for inbound traffic to the intranet
!
prefix, so the neighbor chooses a different path.
default-action reject
!
vpn 2
router
bgp 65040
!
neighbor 40.40.40.100
```

```

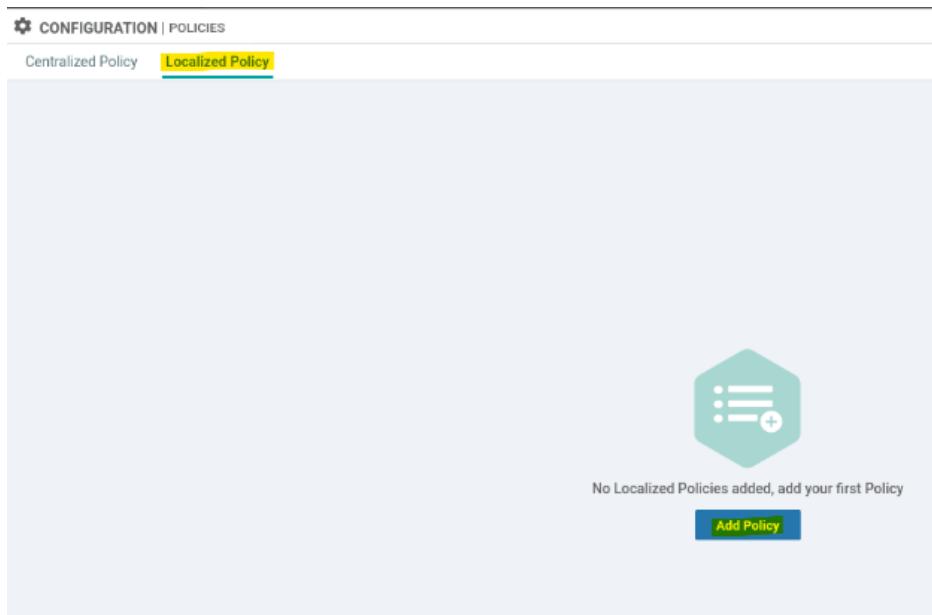
description To_SITE-BANDUNG2
address-family ipv4-unicast
route-policy SET_LOCALPREF_RECEIVED_PREFIX out
!
** If Site-Bandung wants outbound, the local preference to DRC-Cyber 2 is set to 1000, and for DC-DCI it
!
remains at the default.

neighbor 30.30.30.1
description To_vEdge-DC-DCI_eBGP
address-family ipv4-unicast
route-policy DONT_PASSING_USE_OTHER_PATH out
!
** If DC-DCI wants inbound traffic to Site Bandung, the AS-path will be added, causing DC-DCI to choose
!
vEdge-BDG2 (symmetric).

```

// vEdge-BDG2 (via Template)

1. Go to vManage, Configuration > Policies, then select ‘Localized Policies’.

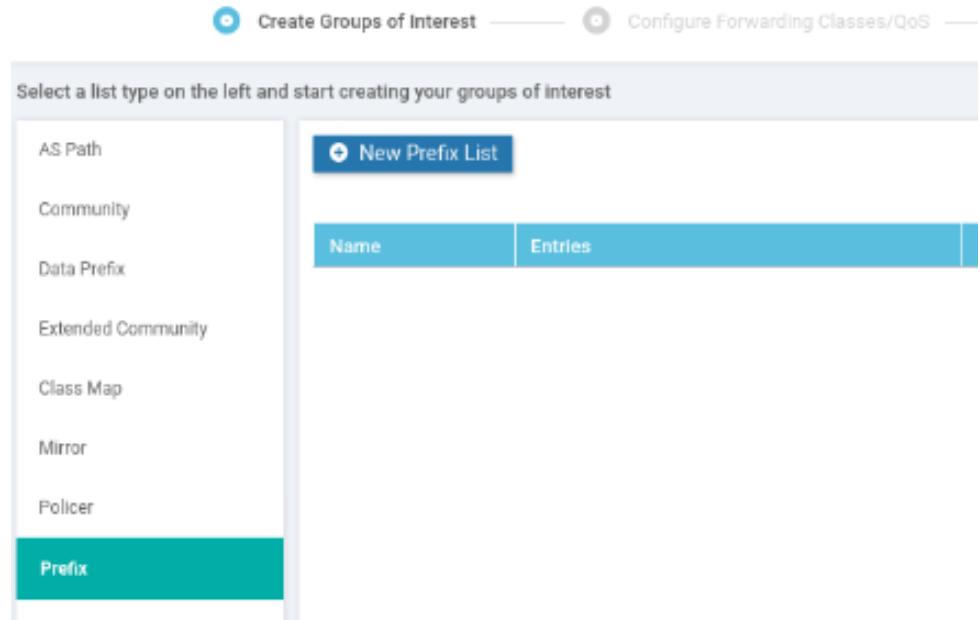


2. There are 5 sections (including overview) for ‘Localized Policies’, where we focus on the ‘Group of Interest (Prefix)’ and ‘Configure Route Policy’ sections only.



3. [Create Groups of Interest]

There are several parameter lists we can use, but here we will use ‘Prefix’. Go to this tab and select ‘New Prefix List’.

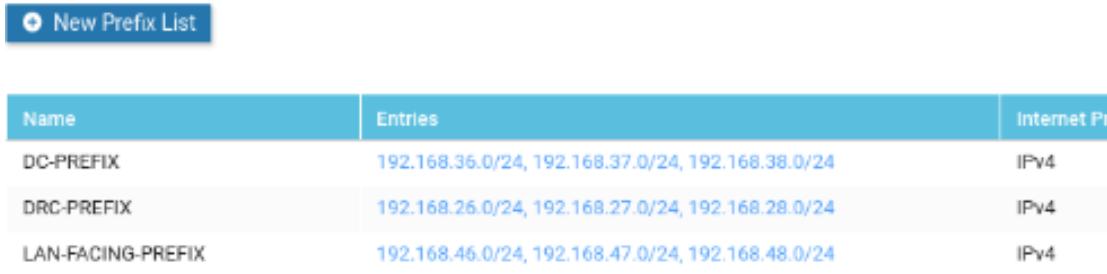


4. Add and categorize prefixes from each site (DC-DCI, DRC-Cyber2, Site-Bandung).

For example, set up a prefix list for the DRC-Cyber2 site.

The screenshot shows the 'New Prefix List' configuration dialog. At the top is a blue header bar with the text 'New Prefix List'. Below it is a section for 'Prefix List Name' containing the value 'DRC-PREFIX'. Under 'Internet Protocol', the 'IPv4' radio button is selected. In the 'Add Prefix' section, the value '192.168.26.0/24, 192.168.27.0/24, 192.168.28.0/24' is listed.

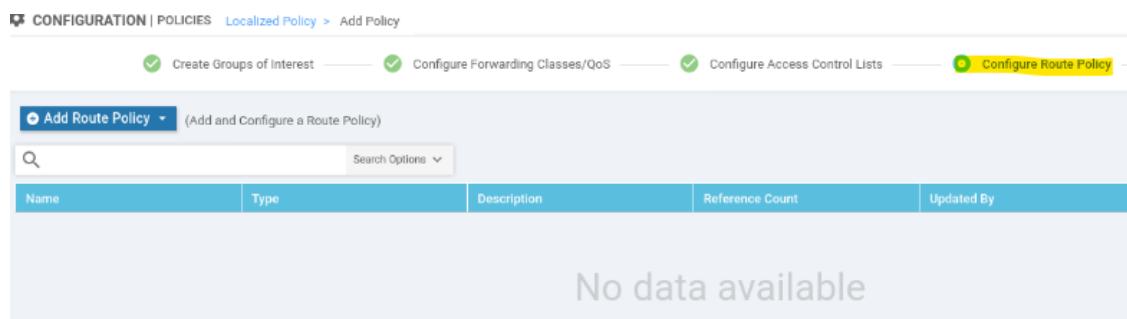
5. Complete all prefix lists from each site that will later be used for ‘match address’ in the route policy configuration



Name	Entries	Internet Pr
DC-PREFIX	192.168.36.0/24, 192.168.37.0/24, 192.168.38.0/24	IPv4
DRC-PREFIX	192.168.26.0/24, 192.168.27.0/24, 192.168.28.0/24	IPv4
LAN-FACING-PREFIX	192.168.46.0/24, 192.168.47.0/24, 192.168.48.0/24	IPv4

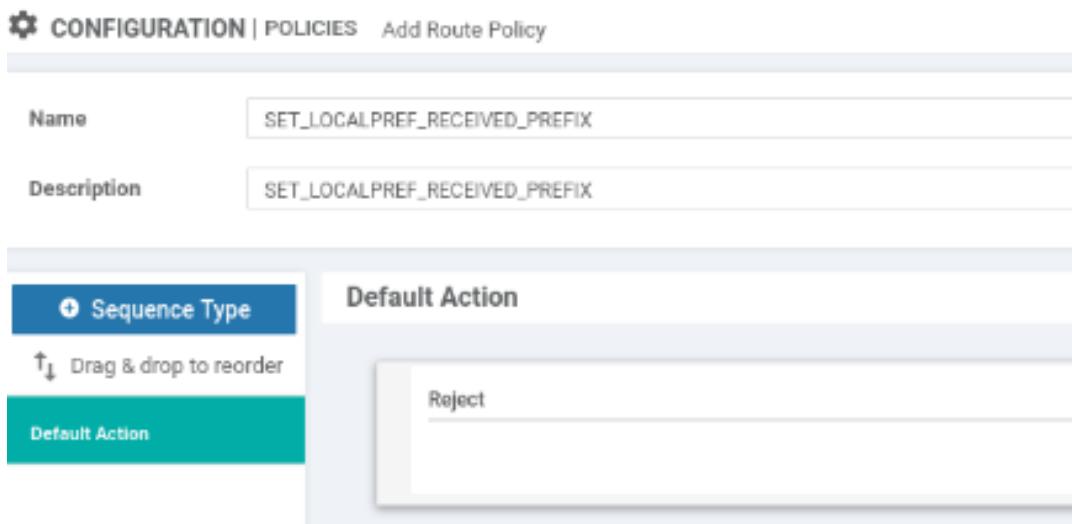
6. [Create Route Policy]

7. Next, go to the next section, skip the ‘QoS’ and ‘ACL’ sections, then in the ‘Route Policy’ section, add a route policy and select ‘Create new’. Here, we’ll create 2 policy ‘SET_LOCALPREF_RECEIVED_PREFIX’ and ‘DONT_PASSING_USE_OTHER_PATH’

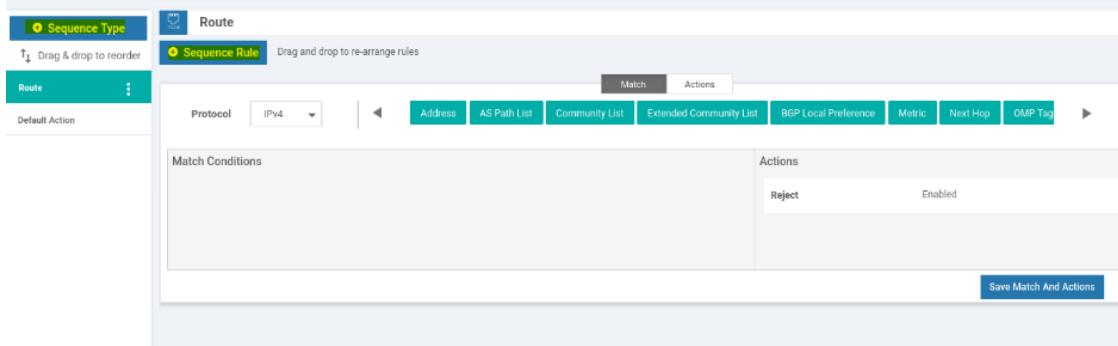


8. ‘SET_LOCALPREF_RECEIVED_PREFIX’

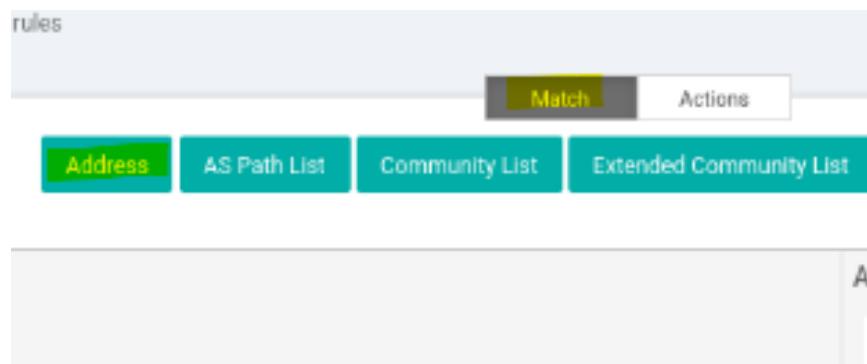
This policy applies to outbound traffic from SITE-BANDUNG2, where it will change the local preference of prefixes received from DC-DCI to 1000 and accept prefixes from DRC-Cyber2 with the default local preference.



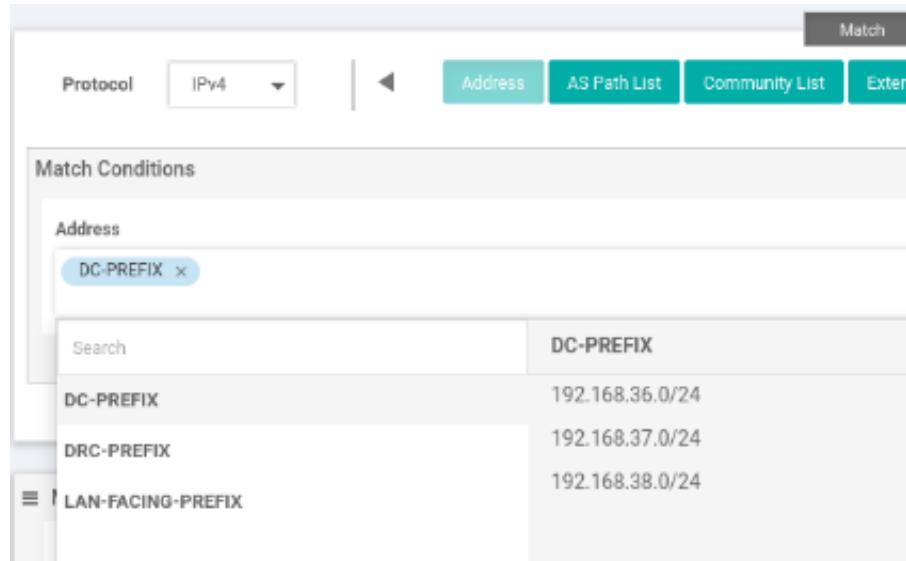
9. Click on ‘Sequence Type’ to add the order of policies to be applied, then click on ‘Sequence Rule’ to add rules to the policy.



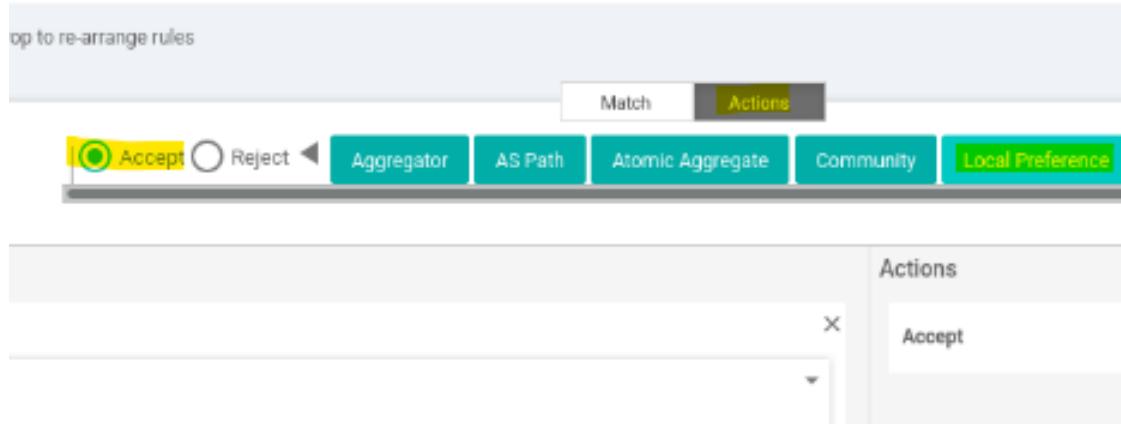
10. In the ‘Match’ menu, select the parameter Address.



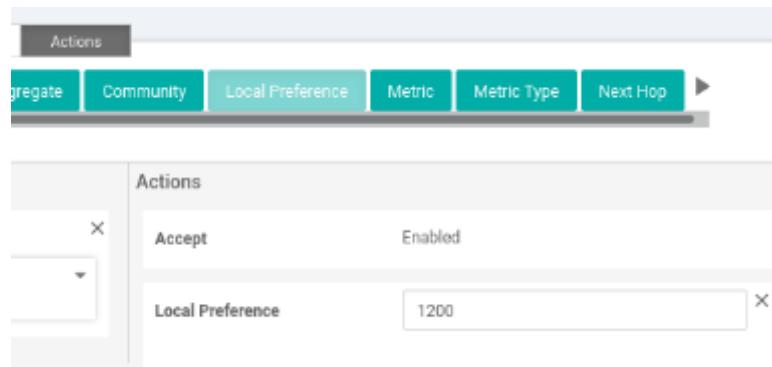
11. As intended, we will match the prefix list for DC-DCI that we will set its local preference.



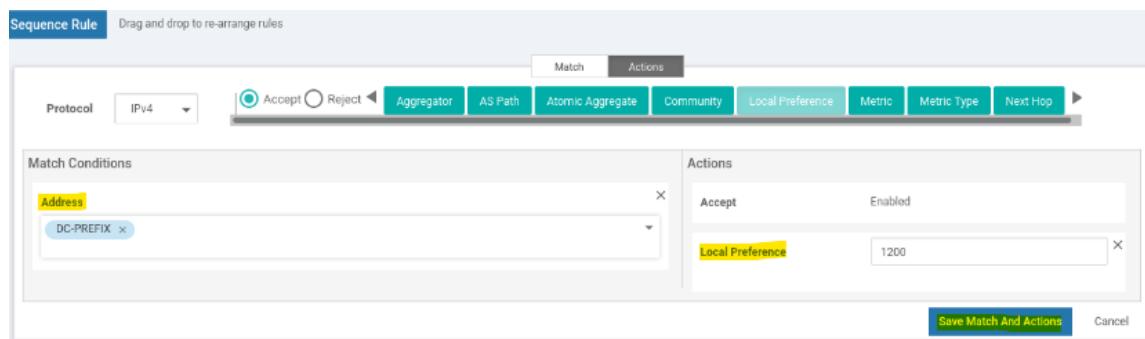
12. Next, move to the ‘Actions’ menu, select ‘Accept’, then select ‘Local Preference’



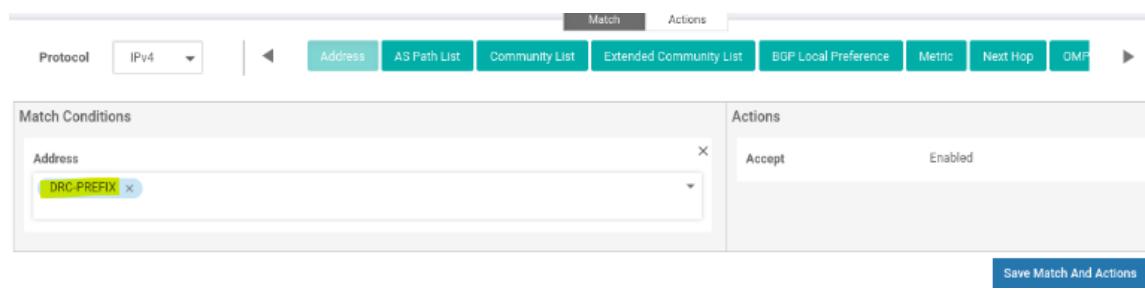
13. Determine the desired local preference value.



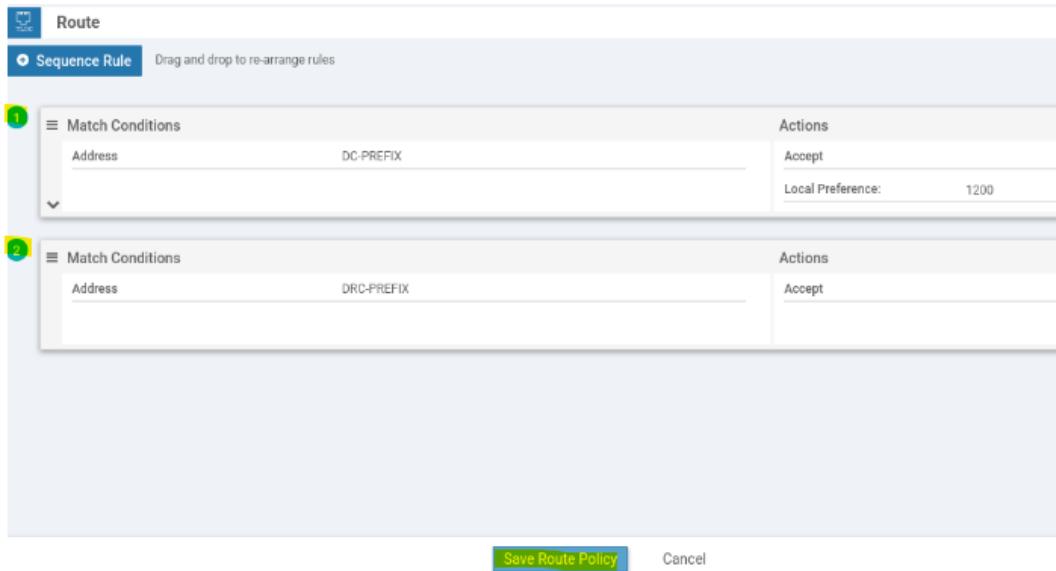
14. Click ‘Save Match and Actions’ when done.



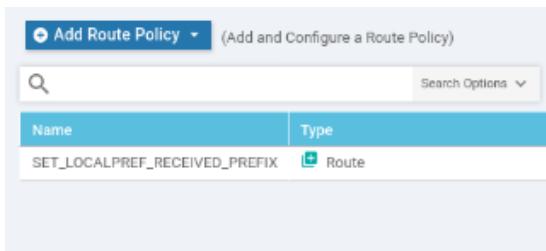
15. Then, add another ‘Sequence Rule’ to match DRC-PREFIX with the action accept without setting a parameter (default local pref).



16. Once the configuration and sequence order are correct, click ‘Save Route Policy’ at the end.

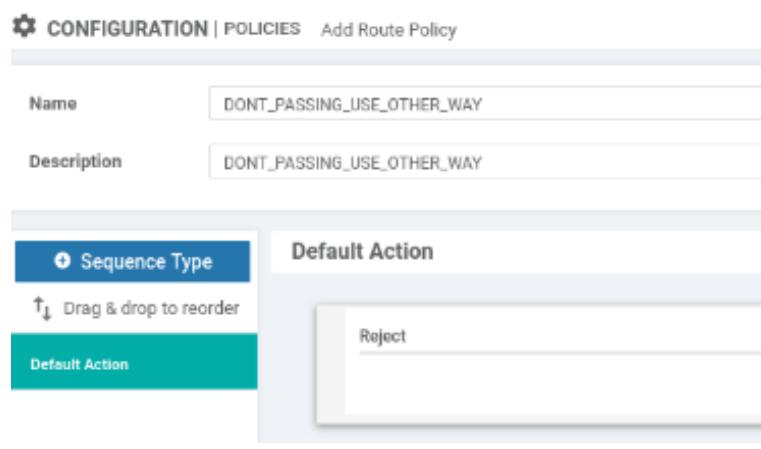


17. Go back to the section, then add the next route policy related to AS-path.

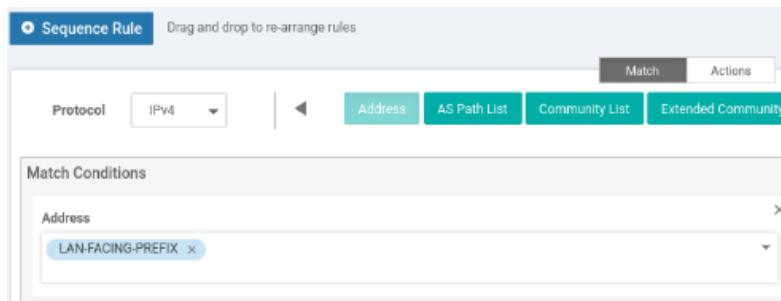


18. ‘DONT_PASSING_USE_OTHER_PATH’

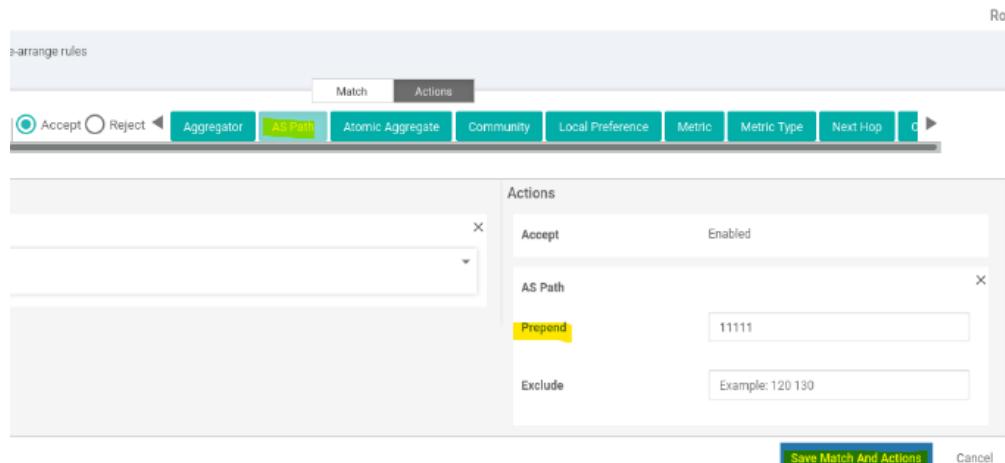
This policy is sent to the neighbor DRC-Cyber2 to inform that to reach the prefix on SITE-BANDUNG2, they must go through the following AS-path (prepend). This makes DRC-Cyber2 choose another path (vEdge-BDG1), which is symmetric with outbound traffic (perspective SITE-BANDUNG2).



19. Add a ‘Sequence Type’ and then a ‘Sequence Rule’. In the ‘Match’ menu, select the prefix list ‘LAN-FACING-PREFIX’ (if DC-DCI is outbound to BANDUNG-SITE2, then the route policy will match).



20. Move to the ‘Actions’ menu, select ‘Accept’, then ‘AS-Path’, and input a fictitious AS-Path. Once done, click ‘Save Match and Actions’ and ‘Save Route Policy’.



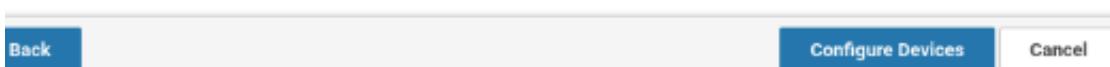
21. Once everything is safe, click ‘Next’ and enter the Policy name to be included in the device template from the Bandung Site.

The screenshot shows the 'Enter name and description for your localized master policy' screen. It has two main sections: 'Policy Name' (BDG-SITE_BGP_ROUTES_POLICY) and 'Policy Description' (BDG-SITE_BGP_ROUTES_POLICY). Below these, there's a 'Policy Settings' section with checkboxes for Netflow, Application, Cloud QoS, Cloud QoS Service side, and Implicit ACL Logging. There's also a 'Log Frequency' field with a placeholder 'Enter in seconds (maximum 2147483647)'. At the bottom, there are buttons for 'BACK', 'Preview', 'Save Policy', and 'CANCEL'.

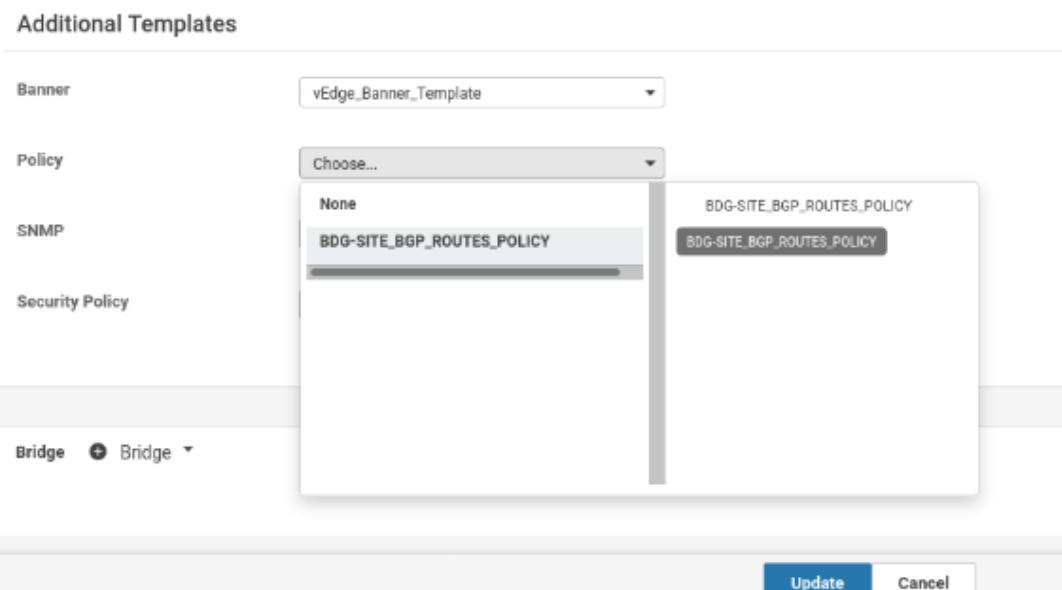
22. Perform a ‘Preview’ of the configuration that will be pushed to the device template.

Once everything is safe, ‘Save Policy’.

```
route-policy SET_LOCALPREF_RECEIVED_PREFIX
sequence 1
match
  address DC-PREFIX
!
action accept
  set
    local-preference 1200
!
!
sequence 11
match
  address DRC-PREFIX
!
action accept
!
!
default-action reject
!
```



23. With this, the policy has been created. Go to Configuration > Templates > Device Template, then edit the Bandung Site Device template, specifically in the ‘Additional Templates’ section, add the created policy, then ‘Update’.



24. Perform the usual push procedure and don’t forget to verify the configuration first. If everything is correct, select ‘Configure Devices’, and the configuration will be successfully pushed to the device.

25. [Apply route policy to BGP and specific neighbor]

26. Finally, apply the route policy that has been created to specific neighbors. Go to the feature template, and edit the feature template Bandung Site VPN 2 BGP, then go to the ‘Neighbor’ section.

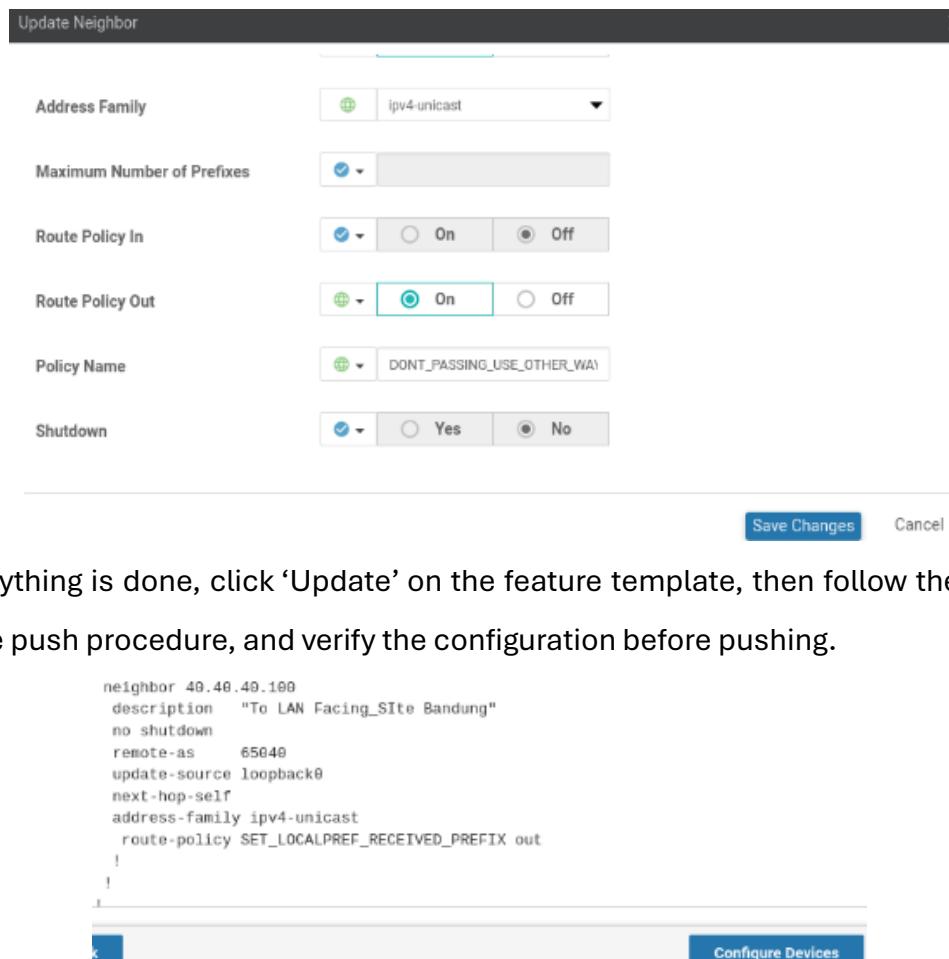
NEIGHBOR			
		IPv4	IPv6
New Neighbor			
Optional		Address	
<input checked="" type="checkbox"/>	[bgp_neighbor_address]	<input type="text"/>	[bgp_neighbor_description]
<input checked="" type="checkbox"/>	[bgp_neighbor_lan_address]	<input type="text"/>	[bgp_neighbor_lan_description]
<input checked="" type="checkbox"/>	[bgp_neighbor_DC_address]	<input type="text"/>	[bgp_neighbor_dc_description]
<input checked="" type="checkbox"/>	[bgp_neighbor_DRC_address]	<input type="text"/>	[bgp_neighbor_DRC_description]

27. Edit the configuration of the SITE-BANDUNG2 neighbor, specifically in the address-family section and apply the policy ‘SET_LOCALPREF_RECEIVED_PREF’ with outbound traffic, then enter the name of route policy. If done, click ‘Save Changes’

Update Neighbor

Address Family	<input type="button" value=""/> <input checked="" type="radio"/> On <input type="radio"/> Off
Address Family	<input type="button" value=""/> <input checked="" type="radio"/> ipv4-unicast
Maximum Number of Prefixes	<input type="button" value=""/> <input checked="" type="radio"/>
Route Policy In	<input type="button" value=""/> <input checked="" type="radio"/> On <input type="radio"/> Off
Route Policy Out	<input type="button" value=""/> <input checked="" type="radio"/> On <input type="radio"/> Off
Policy Name	<input type="button" value=""/> SET_LOCALPREF_RECEIVED_PREF
Protocol	<input type="radio"/> Vpn <input type="radio"/> No
	<input type="button" value=""/> Save Changes <input type="button" value=""/> Cancel

28. Similarly, edit the configuration for the neighbor directed to DRC-Cyber2 and enter the route policy 'DONT_PASSING_USE_OTHER_PATH', choose outbound traffic then enter the name of the route policy. Once done, click 'Save Changes'.



29. If everything is done, click 'Update' on the feature template, then follow the usual device push procedure, and verify the configuration before pushing.

```
neighbor 40.40.40.100
description "To LAN Facing_Site Bandung"
no shutdown
remote-as 65040
update-source loopback0
next-hop-self
address-family ipv4-unicast
route-policy SET_LOCALPREF_RECEIVED_PREFIX out
!
```

30. Dengan demikian route policy via template untuk vEdge-BDG2 berhasil dikonfigurasi

```
vEdge-BDG2# show running-config vpn 2 router bgp neighbor 40.40.40.100
vpn 2
router
bgp 65040
neighbor 40.40.40.100
description "To LAN Facing_Site Bandung"
no shutdown
remote-as 65040
update-source loopback0
next-hop-self
address-family ipv4-unicast
route-policy SET_LOCALPREF_RECEIVED_PREFIX out
!
!
!
!
vEdge-BDG2# show running-config vpn 2 router bgp neighbor 30.30.30.1
vpn 2
router
bgp 65040
neighbor 30.30.30.1
description To_vEdge-DC-DCI_eBGP
no shutdown
remote-as 65030
update-source loopback0
ebgp-multipath 2
address-family ipv4-unicast
!
```

VPN 2 : BGP Route Policy Verification

// show ip bgp (SITE-BANDUNG2)

```
SITE-BANDUNG2#show ip bgp
BGP table version is 46, local router ID is 40.40.40.100
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
               r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
               x best-external, a additional-path, c RIB-compressed,
               t secondary path,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

      Network          Next Hop            Metric LocPrf Weight Path
* i 192.168.26.0    40.40.40.2        11     100      0 65020 i
* >i 192.168.27.0  40.40.40.1        11     1000     0 65020 i
* i 192.168.28.0    40.40.40.2        11     100      0 65020 i
*>i 192.168.36.0    40.40.40.1        11     1000     0 65020 i
* i 192.168.37.0    40.40.40.1        11     100      0 65030 i
*>i 192.168.38.0    40.40.40.1        11     100      0 65030 i
* i 192.168.38.0    40.40.40.2        1200    100      0 65030 i
*>i 192.168.38.0    40.40.40.2        1200    100      0 65030 i
* i 192.168.38.0    40.40.40.1        1200    100      0 65030 i
*>i 192.168.38.0    40.40.40.1        1200    100      0 65030 i
```

DRC-Cyber2 (192.168.2x.0) through 40.40.40.1 (vEdge-BDG1) with local preference 1000

DRC-Cyber2 (192.168.2x.0) through 40.40.40.2 (vEdge-BDG2) with local preference 1200

// show bgp routes (vEdge-DRC-Cyber2 & vEdge-DC-DCI)

```
vEdge-DRC-Cyber2# show bgp routes 192.168.46.0/24
bgp routes-table vpn 2 192.168.46.0/24
  info 0
    nexthop      40.40.40.1
    weight       0
    origin       igp
    as-path      65040
    path-status  valid,best,external
    tag          0
    ospf-tag    0
  info 1
    nexthop      40.40.40.2
    weight       0
    origin       igp
    as-path      "65040 11111"
    path-status  valid,external
    tag          0
    ospf-tag    0
```

vEdge-DRC-Cyber2 choose 40.40.40.1 (vEdge-BDG1) for inbound to SITE-BANDUNG2 (symetric)

```
vEdge-DC-DCI# show bgp routes 192.168.46.0/24
bgp routes-table vpn 2 192.168.46.0/24
info 0
nexthop      40.40.40.1
weight       0
origin       igp
as-path      "65040 11111"
path-status  valid,external
tag          0
ospf-tag     0
info 1
nexthop      40.40.40.2
weight       0
origin       igp
as-path      65040
path-status  valid,best,external
tag          0
ospf-tag     0
```

vEdge-DC-DCI choose 40.40.40.2 (vEdge-BDG2) for inbound to SITE-BANDUNG2 (symetric)

// traceroute (SITE-BANDUNG2)

```
SITE-BANDUNG2#traceroute 192.168.26.10
Type escape sequence to abort.
Tracing the route to 192.168.26.10
VRF info: (vrf in name/id, vrf out name/id)
 1 172.16.42.1 6 msec 21 msec 21 msec
 2 20.20.20.1 51 msec 47 msec 54 msec
 3 172.16.22.10 51 msec 61 msec *
```

Destination to DRC-Cyber2 with next-hop 172.16.42.1 (vEdge-BDG1/40.40.40.1)

```
SITE-BANDUNG2#traceroute 192.168.36.10
Type escape sequence to abort.
Tracing the route to 192.168.36.10
VRF info: (vrf in name/id, vrf out name/id)
 1 172.16.42.2 15 msec 6 msec 20 msec
 2 30.30.30.1 42 msec 48 msec 41 msec
 3 172.16.32.3 68 msec 57 msec *
```

Destination to DC-DCI with next-hop 172.16.42.2 (vEdge-BDG2/40.40.40.2)

VSMART TEMPLATE SETUP AND OVERVIEW

Centralized Policies & Hub and Spoke Setup

Centralized Policies Overview

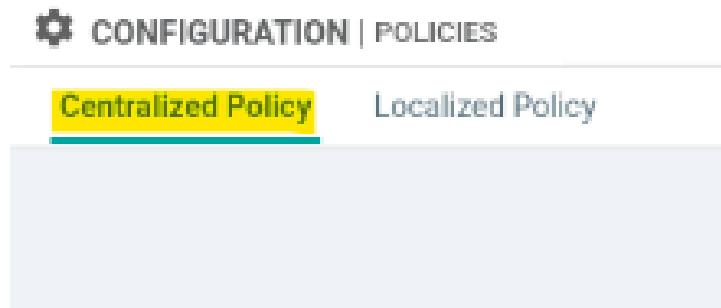
If in the previous BGP route policy we used '**Localized Policies**', there are also '**Centralized Policies**' that can be used for implementation through the vSmart template. The differences between these two methods are as follows:

- **Localized Policies**, Configured directly on the vEdge device itself or locally without affecting other devices. This includes the application of lists, ACL, route policies, QoS, etc.
- **Centralized Policies**, This can be applied as Control and Data Policies, where Control Policies can influence the operation of the next-hop or OMP that forwards traffic, while Data Policies can affect data plane traffic, such as QoS for global vEdge configuration.

Centralized Policies are defined on vManage, then pushed to vSmart, which subsequently pushes them to vEdge. These policies are pushed to all OMP peerings and impact how vEdge handles traffic. When creating control or data policies on vManage, these policies are sent to vSmart. Therefore, it is necessary to set up a device template for vSmart.

What Is Inside Centralized Policies?

In the '**Configuration > Policies**' menu, there are two policy methods: **Centralized Policy** and **Localized Policy**. We will focus on **Centralized Policy**. Click on '**Add Policy**'.



Within **Centralized Policies**, there are four sections that can be configured and applied globally to vEdge devices. These sections include:

a. Create Group of Interest

This section allows us to define several elements to match and apply to policies, such as Applications, Colors, Prefixes, Sites, and more.

 A screenshot of a 'Create Groups of Interest' page. At the top, it says 'CONFIGURATION | POLICIES Centralized Policy > Add Policy'. Below that is a heading 'Create Groups of Interest'. A sub-section title 'Select a list type on the left and start creating your groups of interest' is present. On the left, a vertical list of items under 'Application' includes: Color, Data Prefix, Policer, Prefix, Site, SLA Class, TLOC, and VPN. On the right, a 'New Application List' section shows a table with one row: 'Name' (Microsoft_Apps) and 'Entries' (bing). Another row for Google_Apps is partially visible.

Name	Entries
Microsoft_Apps	bing
Google_Apps	and

b. Configure Topology and VPN Membership

Here, we can control how traffic is forwarded across the backbone and other networks. It includes configurations like Hub and Spoke, Mesh, and Custom Control (Route & TLOC).

The screenshot shows the 'CONFIGURATION | POLICIES' section with 'Centralized Policy > Add Policy'. Below it, there's a 'Create Groups of Interest' button and a 'Configure Topology and VPN Membership' button. A sub-menu titled 'Specify your network topology' is open, showing tabs for 'Topology' (which is selected) and 'VPN Membership'. Under 'Topology', a dropdown menu is open with options: 'Hub-and-Spoke', 'Mesh', 'Custom Control (Route & TLOC)', and 'Import Existing Topology'. To the right of the dropdown is a search bar labeled 'Search Options'. Below the search bar is a table header with columns: Type, Description, and Reference.

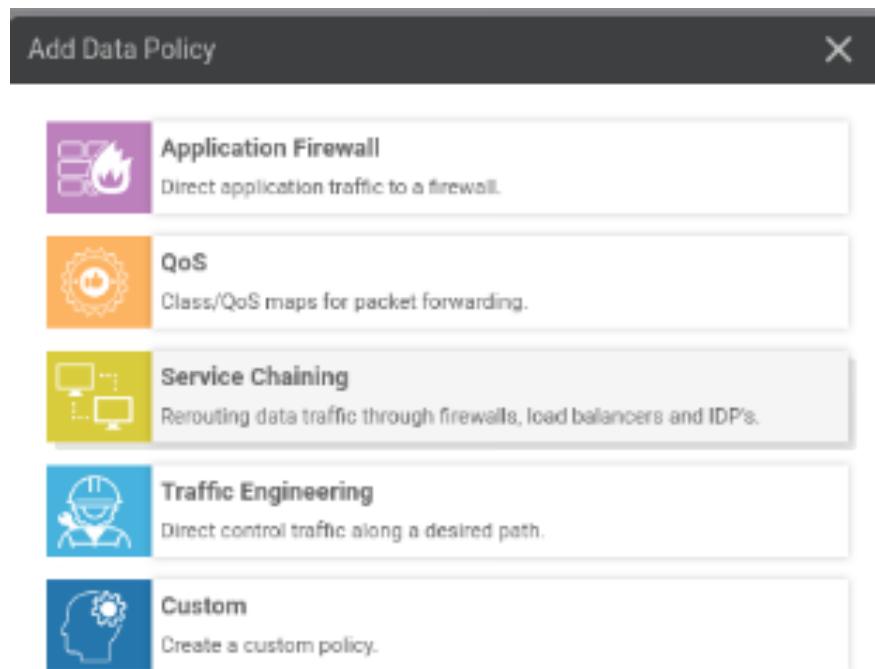
Additionally, in the '**VPN Membership**' tab, we can add or manage policies related to which VPNs are allowed to communicate with each other.

The screenshot shows the same 'CONFIGURATION | POLICIES' section and 'Centralized Policy > Add Policy' path. The 'Create Groups of Interest' and 'Configure Topology and VPN Membership' buttons are present. A sub-menu titled 'Specify your network topology' is open, showing tabs for 'Topology' and 'VPN Membership' (which is selected). Under 'VPN Membership', a section titled 'Add VPN Membership Policy' is shown with the sub-instruction '(Choose and add VPNs to specific site lists)'. Below this is a search bar with a magnifying glass icon and a 'Search Options' dropdown. A table below has columns: Name, Type, Description, and Reference.

c. Configure Traffic Rules

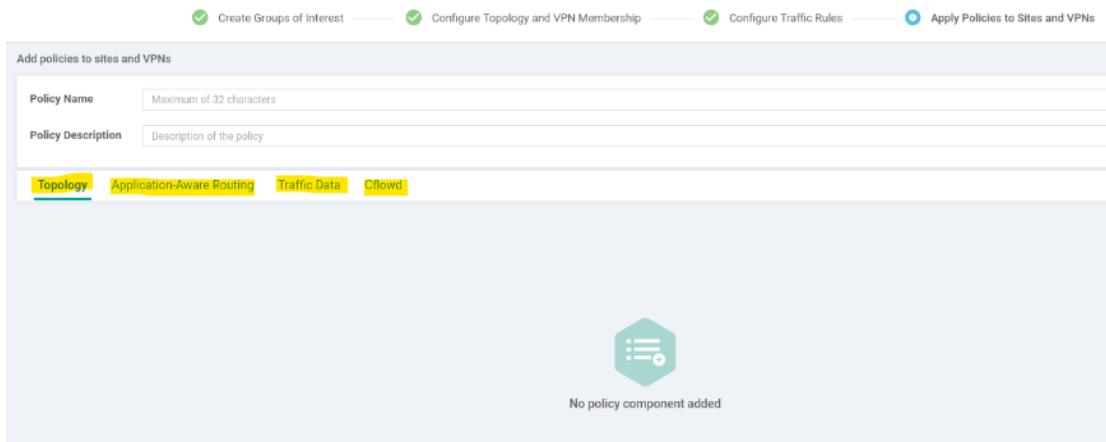
This section allows the use of ‘**Application Aware Routing**’ to monitor network path quality in real time. We can then implement policies for data traffic and configure NetFlow or cFlow as needed.

For instance, in traffic data policies, we can define rules such as redirecting traffic to a firewall, applying traffic engineering, and more.



d. Apply Policies to Sites and VPN's

Finally, this section allows us to review the policies to be implemented. This is where we finalize and ensure the previously defined policies are correctly applied.

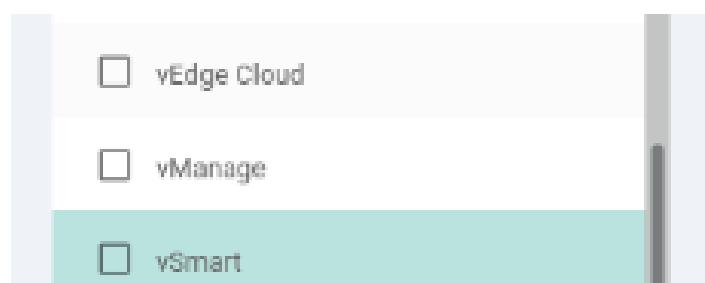


vSmart Template Setup

Create a basic feature template similar to the previous one that will be applied to the device template for vSmart, consisting of:

1. System
2. Banner
3. VPN 0 (Default Route to PE_Gateway)
4. VPN 0 Interface
5. VPN 512
6. VPN 512 Interface

Don't forget to change the device model, where previously we used 'vEdge Cloud,' but for this template, we will use vSmart.



For the vSmart device model, the configurations that can be applied are not as extensive as those available for the vEdge Cloud.

The screenshot shows the 'Feature' tab selected in the top navigation bar. Below it, a breadcrumb navigation shows 'Feature Template > Add Template'. On the left, a sidebar titled 'Select Devices' lists various device models with checkboxes, and the 'vSmart' checkbox is checked and highlighted in green. The main content area is divided into sections: 'BASIC INFORMATION' (containing AAA, Archive, NTP, OMP, Security, System), 'VPN' (containing VPN and VPN Interface Ethernet), and 'OTHER TEMPLATES' (containing Banner, Logging, SNMP). The 'vSmart' template is currently being edited.

Template Setup:

1. [System Template]

The screenshot shows the 'System' configuration page under 'Add Template'. The 'Device Type' is set to 'vSmart', and the 'Template Name' is 'vSmart_System_Template'. The 'Description' field contains the same text. The page is divided into tabs: 'Basic Configuration' (selected), 'GPS', and 'Advanced'. The 'BASIC CONFIGURATION' section contains fields for 'Site ID', 'System IP', 'Overlay ID', 'Hostname', and 'Location'. Each field has a dropdown icon and a placeholder value in brackets: [system_site_id] for Site ID, [system_system_ip] for System IP, '1' for Overlay ID, [system_host_name] for Hostname, and a checked checkbox for Location.

2. [Banner]

Device Type: vSmart
 Template Name: vSmart_Banner_Template
 Description: vSmart_Banner_Template

BASIC CONFIGURATION

Login Banner: Authorized User Only
 MOTD Banner: WELCOME TO MRC ORG

3. [VPN 0]

Feature Template > Add Template > **VPN**

Device Type: vSmart
 Template Name: vSmart_VPN0_Template
 Description: vSmart_VPN0_Template

Basic Configuration DNS IPv4 Route IPv6 Route

BASIC CONFIGURATION

VPN: VPN 0
 Name:

4. Add the IPv4 route with the next-hop to the PE_Router, setting it as a device-specific configuration.

IPV4 ROUTE

New IPv4 Route

Prefix: [vpn0_ipv4_ip_prefix]
 Gateway: Next Hop Null 0 VPN DHCP

Next Hop: 1 Next Hop

5. [VPN 0 Interface]

Feature Template > Add Template > **VPN Interface Ethernet**

Template Name	vSmart_VPN0_Int_Template
Description	vSmart_VPN0_Int_Template
<input checked="" type="button"/> Basic Configuration <input type="button"/> Tunnel <input type="button"/> ARP <input type="button"/> Advanced	
BASIC CONFIGURATION	
Shutdown	<input type="radio"/> Yes <input checked="" type="radio"/> No
Interface Name	<input type="text"/> [vpn0_if_name]
Description	<input type="text"/> [vpn0_if_description]
IP Configuration	
<input type="radio"/> Dynamic <input checked="" type="radio"/> Static	
IPv4 Address	<input type="text"/> [vpn0_if_ip_address]

6. Enable tunnel interface to 'Allow-Service' all

TUNNEL

Tunnel Interface	<input type="radio"/> Off <input checked="" type="radio"/> On <input type="radio"/> On
Color	<input type="color"/> default
Allow Service	
All	<input type="radio"/> Off <input checked="" type="radio"/> On <input type="radio"/> On
DHCP	<input type="radio"/> Off <input checked="" type="radio"/> On <input type="radio"/> On

7. [VPN 512]

Create a feature template for VPN 512 without an IPv4 route.

Feature Template > Add Template > **VPN**

Device Type	vSmart
Template Name	vSmart_VPN512_Template
Description	vSmart_VPN512_Template
<input checked="" type="button"/> Basic Configuration <input type="button"/> DNS <input type="button"/> IPv4 Route <input type="button"/> IPv6 Route	
BASIC CONFIGURATION	
VPN	<input type="radio"/> VPN 512
Name	<input type="text"/>

8. [VPN 512 Interface]

Feature Template > Add Template > [VPN Interface Ethernet](#)

Template Name	vSmart_VPN512_Int_Template
Description	vSmart_VPN512_Int_Template

Basic Configuration Tunnel ARP Advanced

BASIC CONFIGURATION

Shutdown: No

Interface Name: [vpn512_if_name]

Description: [vpn512_if_description]

IP Configuration

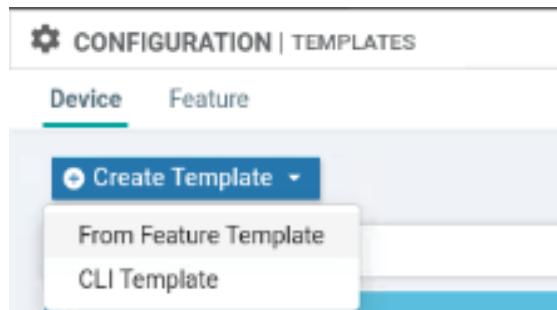
Dynamic Static

IPv4 Address:

9. With the feature template for vSmart ready, it is now ready to be applied to the device template.

Name	Description	Type	Device Model
vSmart_VPN512_Template	vSmart_VPN512_Template	vSmart VPN	vSmart
vSmart_VPN512_Int_Template	vSmart_VPN512_Int_Templ...	vSmart Interface	vSmart
vSmart_VPN0_Template	vSmart_VPN0_Template	vSmart VPN	vSmart
vSmart_VPN0_Int_Template	vSmart_VPN0_Int_Template	vSmart Interface	vSmart
vSmart_System_Template	vSmart_System_Template	vSmart System	vSmart
vSmart_Banner_Template	vSmart_Banner_Template	Banner	vSmart

10. Go to Configuration > Template > Device, then 'Create Template' for the device template.



11. Enter the template name and system feature template.

The screenshot shows the 'Basic Information' tab of a configuration interface. At the top, there are tabs for 'Device' and 'Feature', with 'Device' being the active tab. Below the tabs, there are three input fields: 'Device Model' set to 'vSmart', 'Template Name' set to 'vSmart_Template', and 'Description' set to 'vSmart_Template'. At the bottom of the tab, there are three buttons: 'Basic Information' (highlighted in green), 'Transport & Management VPN', and 'Additional Templates'.

Device Model	vSmart
Template Name	vSmart_Template
Description	vSmart_Template

Basic Information

System *	vSmart_System_Template
Logging*	Factory_Default_Logging_Template

12. In the ‘Transport & Management VPN’ section, also include the VPN 0 and VPN 512 templates that have been created, then ‘Create’.

The screenshot shows the 'Transport & Management VPN' tab of a configuration interface. It contains two sections for 'VPN 0' and 'VPN 512'. Each section has a dropdown for 'VPN 0' and 'VPN Interface'. The 'VPN 0' dropdown is set to 'vSmart_VPN0_Template' and the 'VPN Interface' dropdown is set to 'vSmart_VPN0_Int_Template'. The 'VPN 512' dropdown is set to 'vSmart_VPN512_Template' and the 'VPN Interface' dropdown is set to 'vSmart_VPN512_Int_Template'. Below these sections is an 'Additional Templates' section with dropdowns for 'Banner' (set to 'vSmart_Banner_Template') and 'SNMP' (set to 'Choose...'). At the bottom right are 'Create' and 'Cancel' buttons.

Transport & Management VPN

VPN 0 *	vSmart_VPN0_Template
VPN Interface	vSmart_VPN0_Int_Template

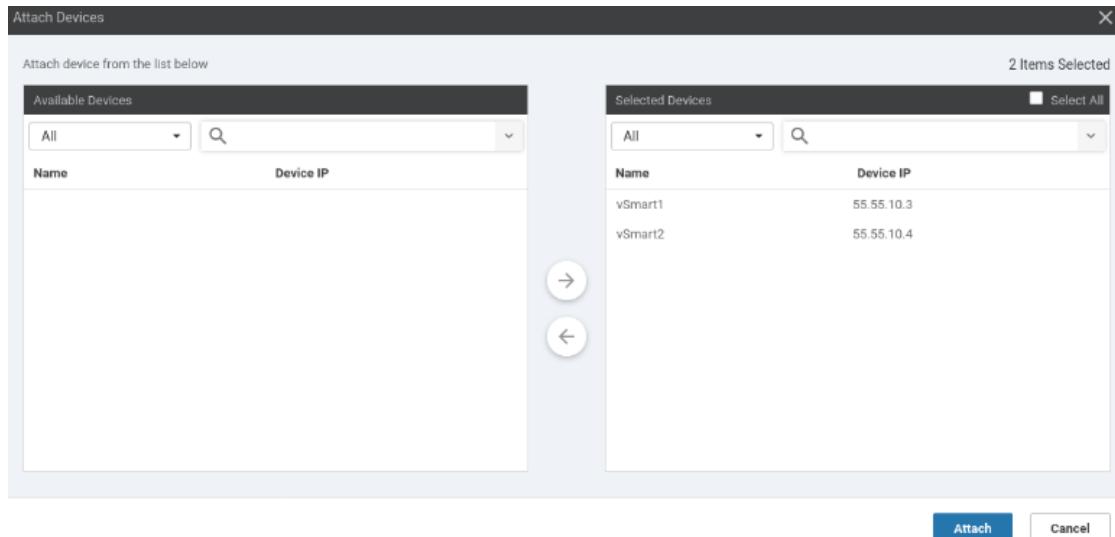
VPN 512 *	vSmart_VPN512_Template
VPN Interface	vSmart_VPN512_Int_Template

Additional Templates

Banner	vSmart_Banner_Template
SNMP	Choose...

Create **Cancel**

13. Attach the device to the device template by moving the vSmart to the right, then ‘Attach’.



14. Edit both vSmart devices according to the required parameter values, then ‘Update’ and ‘Next’.

Update Device Template

Variable List (Hover over each field for more information)

Chassis Number	c783a5bf-7e81-4ecc-a82d-e243db67a18c
System IP	55.55.55.3
Hostname	vSmart1
Interface Name(vpn512_if_name)	eth0
Description(vpn512_if_description)	To MGMT Interface
Prefix(vpn0_ipv4_ip_prefix)	0.0.0.0/0
Address(vpn0_next_hop_ip)	100.10.1.10
Interface Name(vpn0_if_name)	eth1
Description(vpn0_if_description)	To PE_Router gateway
IPv4 Address(vpn0_if_ip_address)	100.10.1.13/24
Hostname(system_host_name)	vSmart1
System IP(system_system_ip)	55.55.55.3
Site ID(system_site_id)	10

Generate Password **Update** **Cancel**

15. Verify the configuration before pushing it to the devices. Once it is safe, select ‘Configure Devices’.

```

:
vpn 0
interface eth1
description "To PE_Router gateway"
ip address 100.10.1.13/24
tunnel-interface
allow-service all
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
!
no shutdown
!
ip route 0.0.0.0/0 10.100.1.10
!
```

Back **Configure Devices** **Cancel**

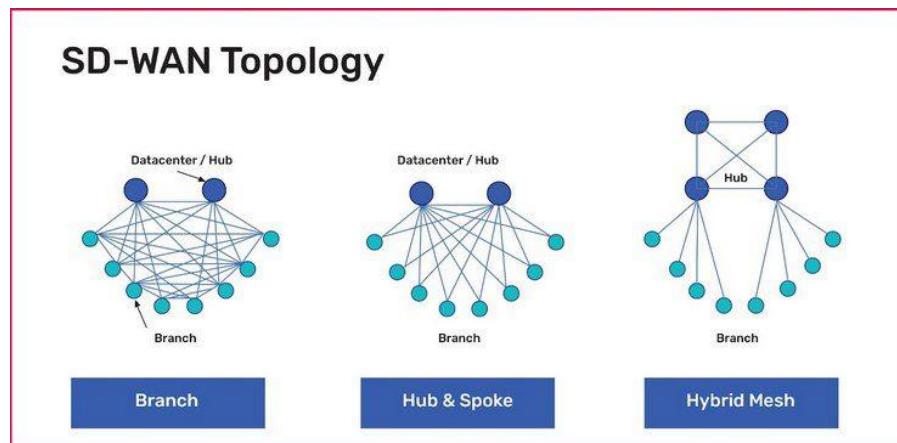
16. As a result, the configuration from the device template is successfully pushed to vSmart, and both vSmart devices are now managed by vManage. Now, vSmart can be configured with Centralized Policies.

```

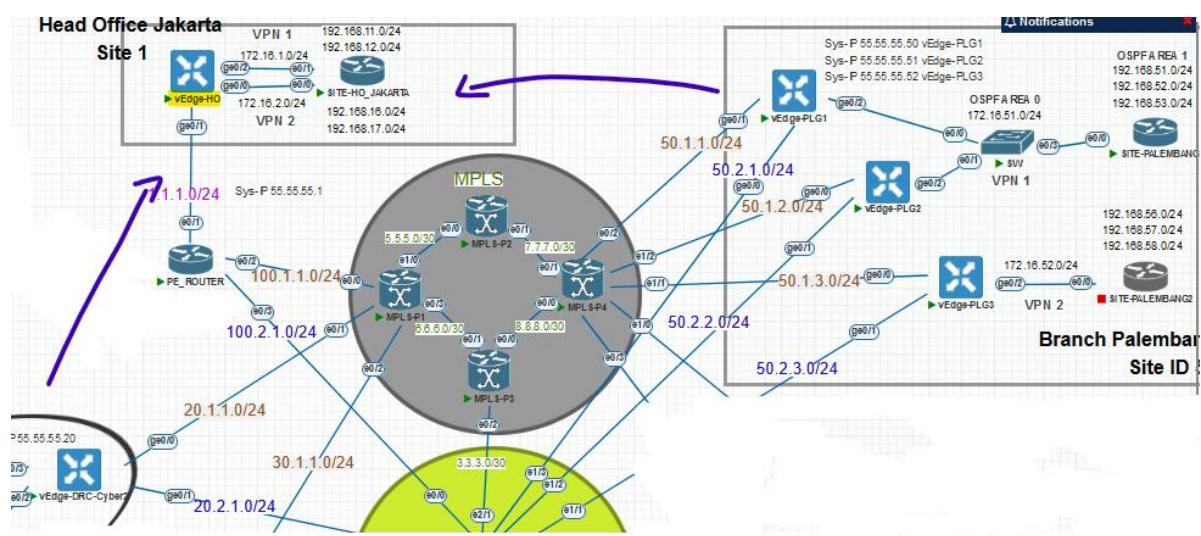
vSmart1# show run vpn 0
vpn 0
interface eth1
description "To PE_Router gateway"
ip address 100.10.1.13/24
tunnel-interface
allow-service all
allow-service dhcp
allow-service dns
allow-service icmp
no allow-service sshd
no allow-service netconf
no allow-service ntp
no allow-service stun
!
no shutdown
!
ip route 0.0.0.0/0 100.10.1.10
!
```

Hub & Spoke Setup

If previously, the connection between vEdge was established with a ‘Spoke-to-Spoke’ or Branch-to-Branch model through IPSec tunnels, with the application of Centralized Policies, we can control each traffic between vEdges to become ‘Hub & Spoke’. This type of model is a network architecture used to connect various locations or branches to a central point (Hub). Therefore, all communications between vEdge (branch-to-branch) must go through the Hub (head office).



Skenario:



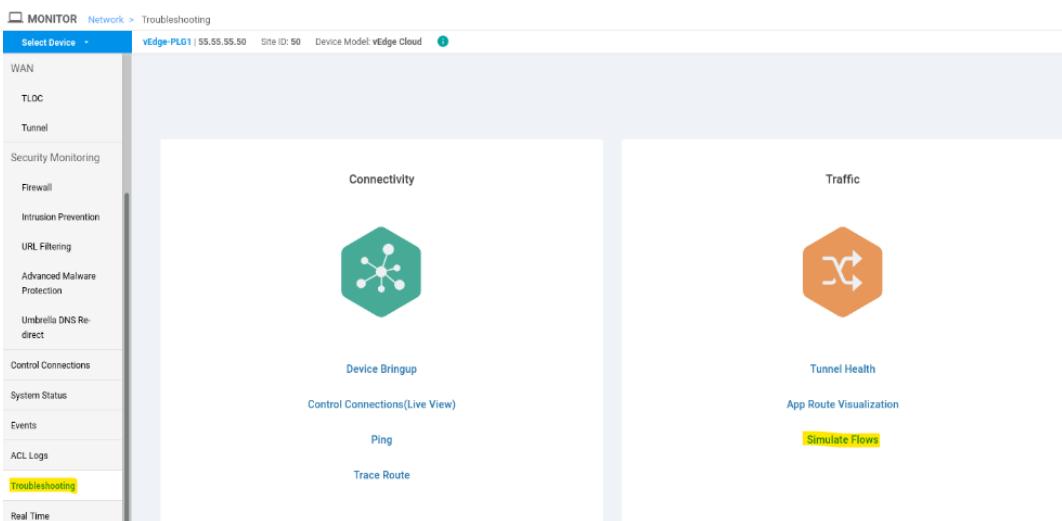
If by default, each vEdge formed direct communication with each other (full-mesh), here we will configure that each vEdge (Spoke) must go through the Hub first before communicating with another vEdge.

Here, **vEdge-HO** will act as the **Hub**, and all vEdge at **Palembang-Site** and **vEdge-DRC-Cyber2** will be **Spokes**.

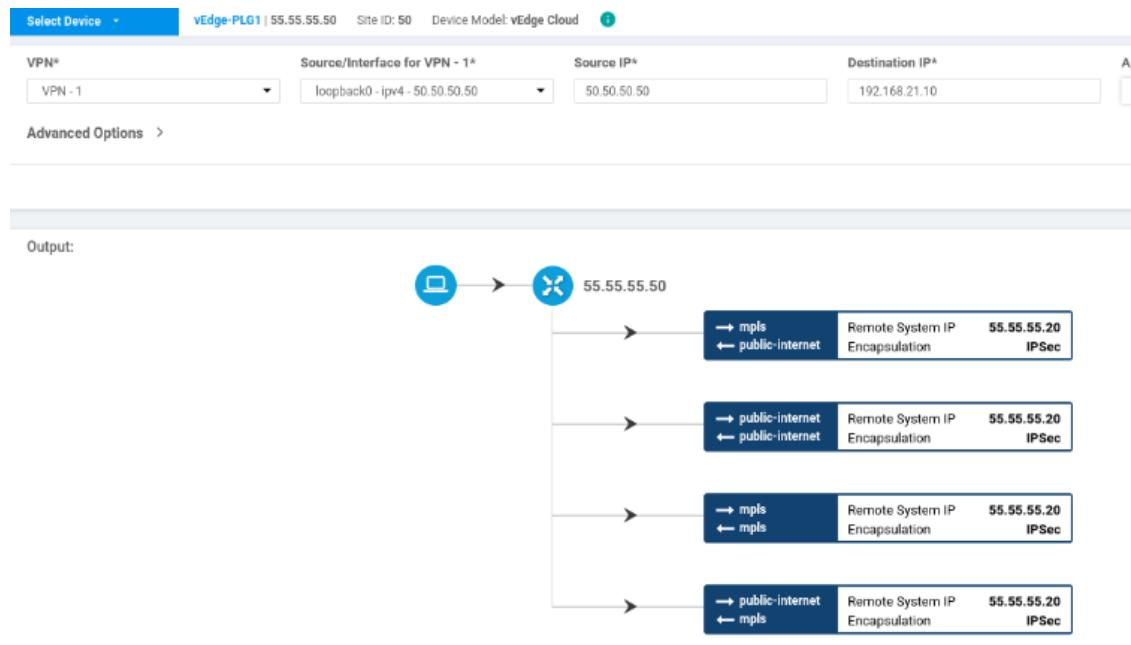
Step-by-step:

// Before scenario

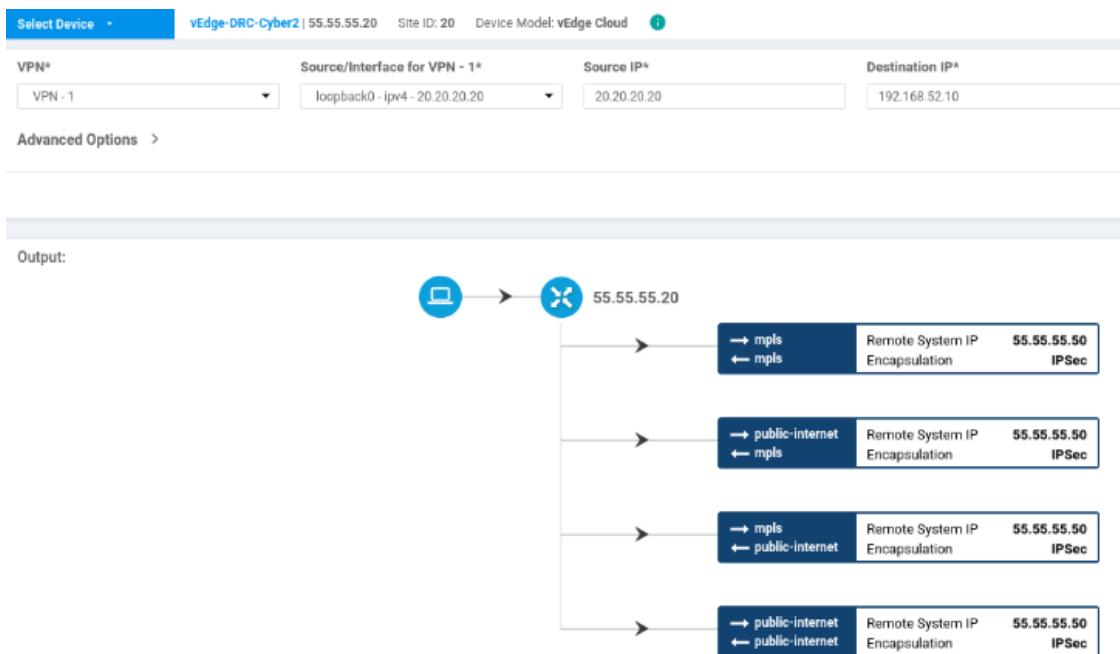
1. We will perform a traceroute from vEdge-PLG to vEdge-DRC-Cyber2. In vManage, go to Monitor > Network, select the device, then choose the Troubleshooting menu, and finally ‘Simulate Flows’.



2. Specify the VPN, Source Interface/IP, and Destination IP, then click ‘Simulate’. This allows us to directly see the traffic flow to vEdge-DRC-Cyber2 (Spoke).



3. Repeat the same from the perspective of vEdge-DRC-Cyber.



4. We will configure all traffic between sites to first pass through vEdge-HO (HUB).

// Configuring Hub & Spoke Template

1. Go to Configuration > Centralized Policy, then 'Add Policy'.

The screenshot shows the 'CONFIGURATION | POLICIES' screen under the 'Centralized Policy' tab. The interface includes a navigation bar with 'Centralized Policy' and 'Localized Policy' tabs. A large central area displays a green hexagonal icon with a list and a plus sign, indicating no policies have been added. Below the icon is the text 'No Centralized Policies added, add your first Policy'. At the bottom right is a prominent blue 'Add Policy' button.

2. In the ‘Group of Interest’ section, we will add parameters such as Site, TLOC, and VPN.

Select a list type on the left and start creating your groups of interest

Name	Entries	Reference Count
Microsoft_Apps	bing, hockeyapp, live_hotmail, ly...	0
Google_Apps	android-updates, blogger, chro...	0

3. [Group of Interest – Site]

Add ‘New Site List’ and designate Site HO as Hub, and Sites PLG and DRC as spokes.

Click ‘Add’.

Name	Entries	Reference Count	Updated By	Last Updated	Action
HO-HUB	1	0	admin	18 Dec 2024 10:27:52 AM GMT	

4. [Group of Interest – TLOC]

Add ‘New TLOC List’ and input the TLOC information from vEdge-HO (Hub). The TLOC IP should match the system IP of the device, adjust the Color, and set the Encapsulation to IPsec. Then, click ‘Save’.

The screenshot shows a dialog box titled "TLOC List". It has a "List Name" field containing "HUB-HO-TLOC". Below it is a table with four columns: "TLOC IP" (containing "50.50.50.1"), "Color" (containing "metro-ethernet"), "Encap" (containing "ipsec"), and "Preference" (containing "0-4294967295"). A blue "Add TLOC" button is located below the table. At the bottom right are "Save" and "Cancel" buttons.

5. [Group of Interest – VPN]

Add the relevant VPN list operating.

The screenshot shows a dialog box for "New VPN List". It has a "VPN List Name" field with "VPN1" entered. Below it is a "Add VPN" section with a "2" entry and an "Add" button. At the bottom is a table with one row, showing "Name: VPNT1", "Entries: 1", "Reference Count: 1", "Updated By: admin", and "Last Updated: 18 Dec 2024 9:26:48 AM GMT".

6. Then, click ‘Next’. In the ‘Configure Topology and VPN Membership’ section, add ‘Topology’ and select ‘Hub & Spoke’.

The screenshot shows the "Configure Topology and VPN Membership" section. At the top, there are two radio buttons: "Create Groups of Interest" (checked) and "Configure Topology and VPN Membership" (unchecked). Below is a "Specify your network topology" area. Under "Topology", the "Topology" tab is selected, and the "Add Topology" button is highlighted. A dropdown menu shows "Hub-and-Spoke" (selected), "Mesh", "Custom Control (Route & TLOC)", and "Import Existing Topology". To the right is a search bar and a table with columns "Type", "Description", and "Reference".

7. Name and describe the policy being created, then select the VPN list to be applied.

CONFIGURATION POLICIES		Add Hub-and-Spoke Policy
Name	HUB-SPOKE-PLG-DRC	
Description	YOU NEED TO GO TO HO FIRST	
VPN List	VPN1	

8. In the 'Add Hub & Spoke Sites' column, add the Site-list we created, for Hub Sites direct to HO-HUB, then 'Add'.

Add Hub-and-Spoke Sites

Add Hub Sites

Site List

HO-HUB

Search

HO-HUB

PLG-DRC-SPOKE

9. For Spoke-Sites, direct to PLG-DRC-SPOKE, then 'Add'.

Add Spoke Sites

Site List

PLG-DRC-SPOKE

Action

Add Cancel

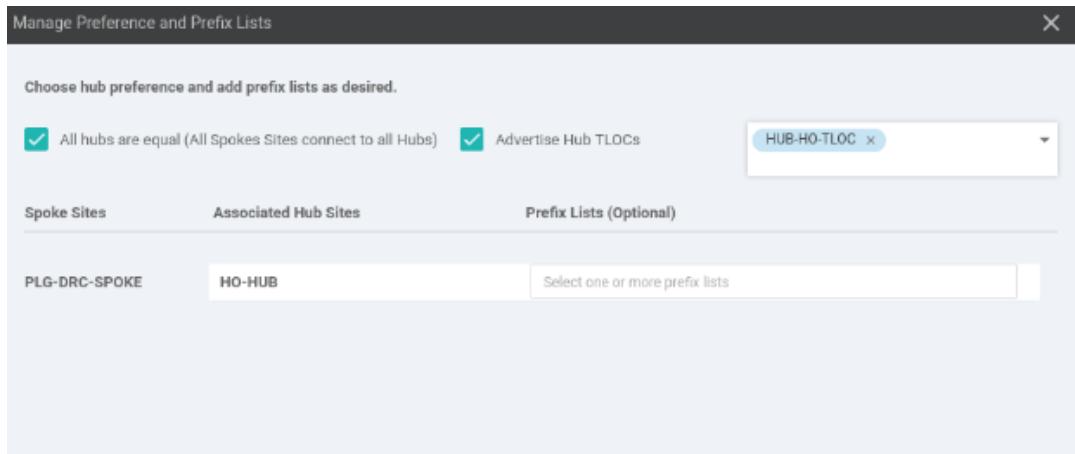
Site Lists

10. Then, a setup for 'Hub Preferences' will appear, click on the option.

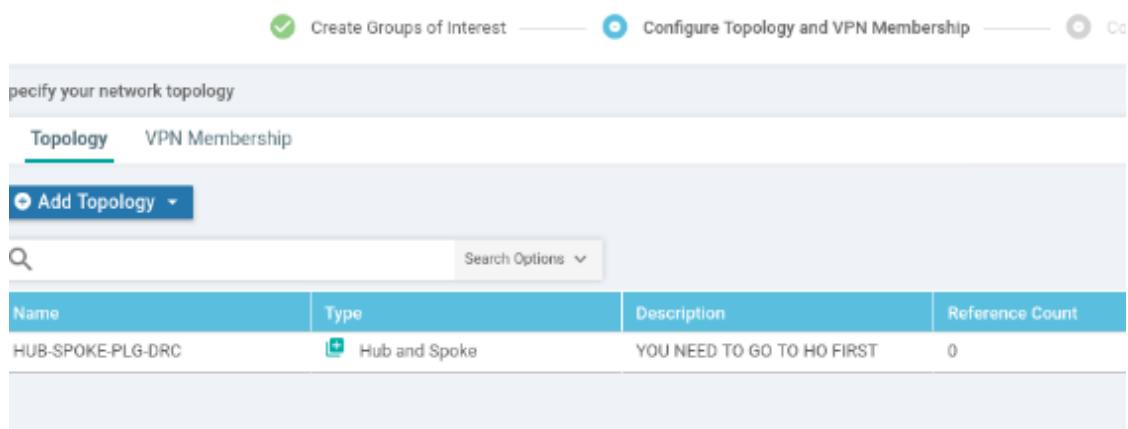
Hub Preferences

Manage Custom Preferences and Prefix Lists

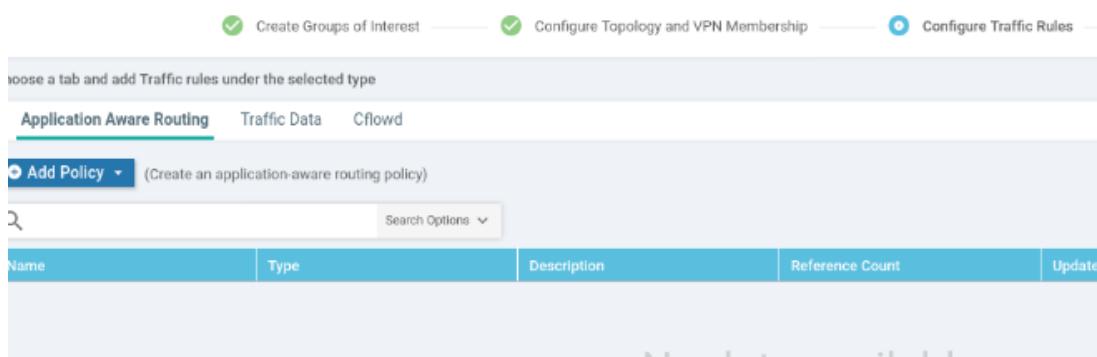
11. Check ‘Advertise Hub TLOC’, then direct to ‘HUB-HO-TLOC’ that was created, then ‘Save Changes’.



12. Once complete, ‘Save Hub & Spoke Policy’. The policy has now been successfully created.



13. Click ‘Next’, skip the ‘Configure Traffic Rules’ section, and then click ‘Next’ again.



14. Lastly, in the ‘Apply Policies to Sites and VPN’, input the name and description of the policy.

15. Perform a ‘Preview’ to confirm the configuration before pushing it to vSmart. The ‘default-action reject’ ensures that every spoke must go through the hub first. Once verified, click ‘Save Policy’.

```

policy
control-policy control_1877328299
sequence 10
match route
site-list HO-HUB
vpn-list VPN1
!
action accept
!
sequence 20
match route
site-list PLG-DRC-SPOKE
vpn-list VPN1
!
action accept
set
tioc-list HUB-HO-TLOC
!
!
sequence 30
match tioc
site-list HO-HUB
!
action accept
!
!
default-action reject
!
lists

```

Save Policy **BACK**

16. Activate the policy that was created, and thus the Hub & Spoke configuration will be pushed to vSmart.

17. The Hub and Spoke configuration has been successfully pushed to vSmart.

```
vSmart1# show running-config policy
policy
  lists
    vpn-list VPN1
      vpn 1
    !
    tloc-list HUB-HO-TLOC
      tloc 50.50.50.1 color metro-ethernet encaps ipsec
    !
    site-list HO-HUB
      site-id 1
    !
    site-list PLG-DRC-SPOKE
      site-id 20
      site-id 50
    !
    !
    control-policy control_1877328209
      sequence 10
        match route
          site-list HO-HUB
          vpn-list VPN1
        !
        action accept
      !
    !
  !
!
```

// After Scenario (Unreachable Spoke Solved)

After implementing Hub & Spoke, there is an issue where spoke devices only have OMP routes from the Hub, resulting in a problem with reachability to prefixes behind the spoke/vEdge devices.

VPN	PREFIX	PROTOCOL	PROTOCOL	NEXTHOP SUB TYPE	IF NAME	NEXTHOP ADDR	NEXTHOP VPN	TLOC IP	COLOR
1	20.20.20.20/32	ospf	IA	loopback0	-	-	-	-	-
1	20.20.20.20/32	connected	-	loopback0	-	-	-	-	-
1	35.35.35.1/32	omp	-	-	-	-	55.55.55.1	metro-et	
1	172.16.1.0/24	omp	-	-	-	-	55.55.55.1	metro-et	
1	172.16.21.0/24	ospf	IA	ge0/3	-	-	-	-	-
1	172.16.21.0/24	connected	-	ge0/3	-	-	-	-	-
1	192.168.11.0/24	omp	-	-	-	-	55.55.55.1	metro-et	
1	192.168.12.0/24	omp	-	-	-	-	55.55.55.1	metro-et	
1	192.168.13.0/24	omp	-	-	-	-	55.55.55.1	metro-et	
1	192.168.21.0/24	ospf	IA	ge0/3	172.16.21.10	-	-	-	-
1	192.168.22.0/24	ospf	IA	ge0/3	172.16.21.10	-	-	-	-
1	192.168.23.0/24	ospf	IA	ge0/3	172.16.21.10	-	-	-	-

```
vEdge-DRC-Cyber2(config-vpn-1)# do ping vpn 1 192.168.51.10
Ping in VPN 1
PING 192.168.51.10 (192.168.51.10) 56(84) bytes of data.
From 127.1.0.2 icmp_seq=1 Destination Net Unreachable
From 127.1.0.2 icmp_seq=2 Destination Net Unreachable
```

The solution to this is to **add a default route** on the vEdge-HO (Hub) with a **next-hop of 'null0'**, then advertise this default as an OMP (omp advertise static). This is used for routing simplification and to avoid routing loops. In short, every spoke will have a default route to the Hub. When a spoke wants to send packets to an unknown route, that route will be discarded (black hole) to prevent routing loops (going back and forth between the hub and spoke).

Config:

```
vEdge-HO(config)# vpn 1
vEdge-HO (config-vpn-1)# ip route 0.0.0.0/0 null0
vEdge-HO (config-vpn-1)# omp
vEdge-HO (config-omp)# advertise static
vEdge-HO (config-omp)# commit
```

Commit complete.

So, after vEdge-HO (Hub) redistributes its default route to OMP, every spoke now has a default route going to the Hub, and reachability to Site 20 is now connected.

VPN	PREFIX	PROTOCOL	PROTOCOL SUB TYPE	NEXTHOP IF NAME	NEXTHOP ADDR	NEXTHOP VPN	TLOC IP
1	0.0.0.0/0	omp	-	-	-	-	55.55.55.1
1	20.20.20.20/32	ospf	IA	loopback0	-	-	-
1	20.20.20.20/32	connected	-	loopback0	-	-	-
1	35.35.35.1/32	omp	-	-	-	-	55.55.55.1

^Aborted: by user

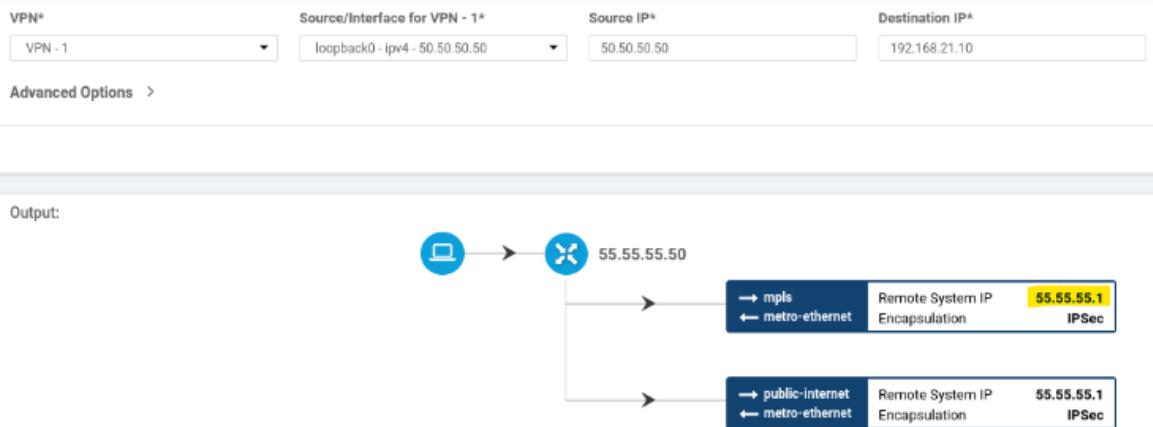
```
vEdge-DRC-Cyber2(config-vpn-1)# do ping vpn 1 192.168.51.10
Ping in VPN 1
PING 192.168.51.10 (192.168.51.10) 56(84) bytes of data.
64 bytes from 192.168.51.10: icmp_seq=1 ttl=253 time=70.3 ms
64 bytes from 192.168.51.10: icmp_seq=2 ttl=253 time=87.8 ms
64 bytes from 192.168.51.10: icmp_seq=3 ttl=253 time=86.9 ms
64 bytes from 192.168.51.10: icmp_seq=4 ttl=253 time=87.7 ms
```

VPN	PREFIX	PROTOCOL	PROTOCOL SUB TYPE	NEXTHOP IF NAME	NEXTHOP ADDR	NEXTHOP VPN	TLOC IP
1	0.0.0.0/0	omp	-	-	-	-	55.55.55.1
1	35.35.35.1/32	omp	-	-	-	-	55.55.55.1
1	50.50.50.50/32	ospf	IA	loopback0	-	-	-
1	50.50.50.50/32	connected	-	loopback0	-	-	-

^Aborted: by user

```
vEdge-PLG1(config-vpn-1)# do ping vpn 1 192.168.22.10
Ping in VPN 1
PING 192.168.22.10 (192.168.22.10) 56(84) bytes of data.
64 bytes from 192.168.22.10: icmp_seq=1 ttl=253 time=71.4 ms
64 bytes from 192.168.22.10: icmp_seq=2 ttl=253 time=73.8 ms
64 bytes from 192.168.22.10: icmp_seq=3 ttl=253 time=73.6 ms
```

// Traffic flows



Traffic is first directed to vEdge-HO/Hub.

// Traceroute



REFERENCES

- https://www.youtube.com/watch?v=do_Tcdyw6vo&list=PLxyr0C_3Ton1mWNeKEnDtIgqZS_fQKQyL
- https://www.youtube.com/watch?v=LA2iZCX76sl&list=PLpfp_h7oipaS45E1JYvuBi0ypzJOKhJ1c
- <https://www.youtube.com/watch?v=RxQPqmHm6YU&t=2032s>
- <https://www.networkacademy.io/ccie-enterprise/sdwan>