

CISCO *Live!*

Let's go



The bridge to possible

Empowering Your Network with SDWAN OMP:

Path Optimization and Policy Insights use cases

Waqas Daar, Customer Success Specialist SDWAN/Thousand Eyes

CISCO Live!

BRKENT-3115

“If you cannot explain it simply, you don’t understand it well enough.”

Albert Einstein

Who am I?



- Master of Science (MSc.) in Internetworking from KTH Royal Institute of Technology, Stockholm Sweden.
- Started working at Cisco Systems Belgium as a Technical Assistance Center (TAC) Engineer in 2011.
- CCIE# 42704 (ENT,SP)

Baseline and Objectives

- The session main objectives are:
 - To have better understanding about OMP, OMP peering, OMP best path selection, OMP loop avoidance in SD-WAN fabric etc..
 - To understand how RIB-IN/RIB-OUT are calculated on SD-WAN controller and how to address it from scale perspective.
 - To have better understanding how Multicast tree is built in SD-WAN fabric.
- Cisco SD-WAN basic level knowledge is a must
- This is advanced level session, so basics are not covered
- The session **does not cover** any centralized data policies, Application Aware Routing (AAR), QoS policies, SD-WAN Multi-Region Fabric (MRF) etc...

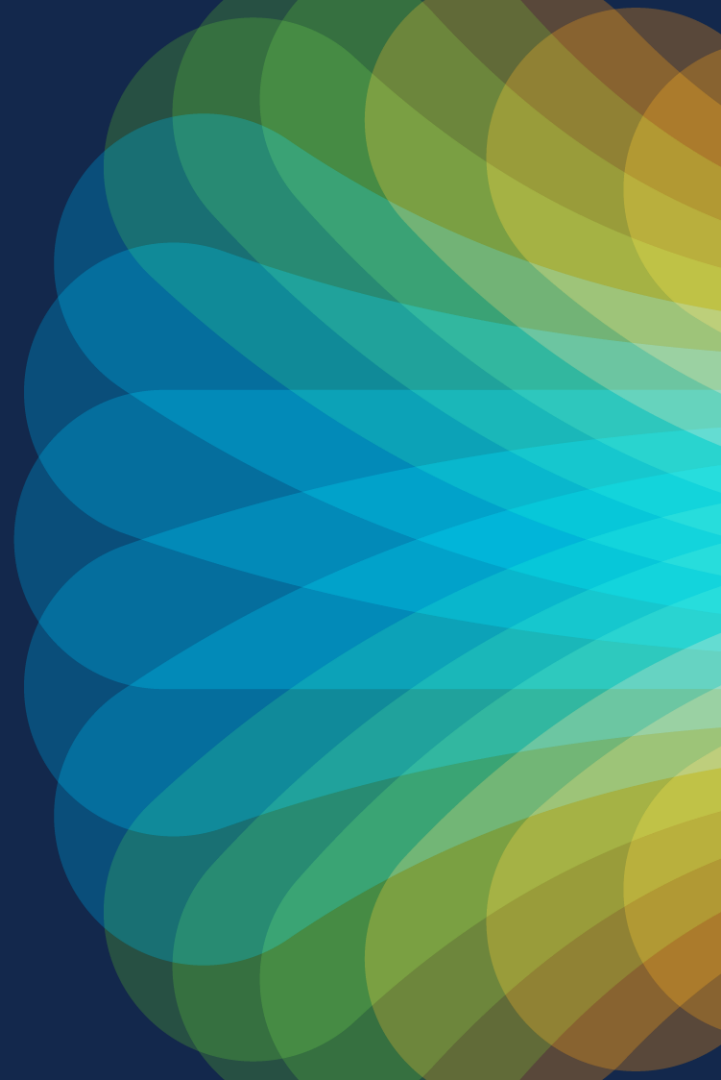
Agenda

- Part 1: SD-WAN OMP Peering Deep dive
 - Intro and OMP Quick overview
 - Demystifying OMP Peering
 - OMP Packet Insights and commonalities with BGP
 - Walkthrough of OMP Session establishment from the WAN Edge Process level
 - Navigating Through Logs During OMP Session Establishment
 - Troubleshooting OMP Peering
- Part 2: SD-WAN controller OMP deep dive
 - SD-WAN controller RIB-IN/RIB-OUT calculation
 - OMP Best path selection process how OMP loop avoidance works

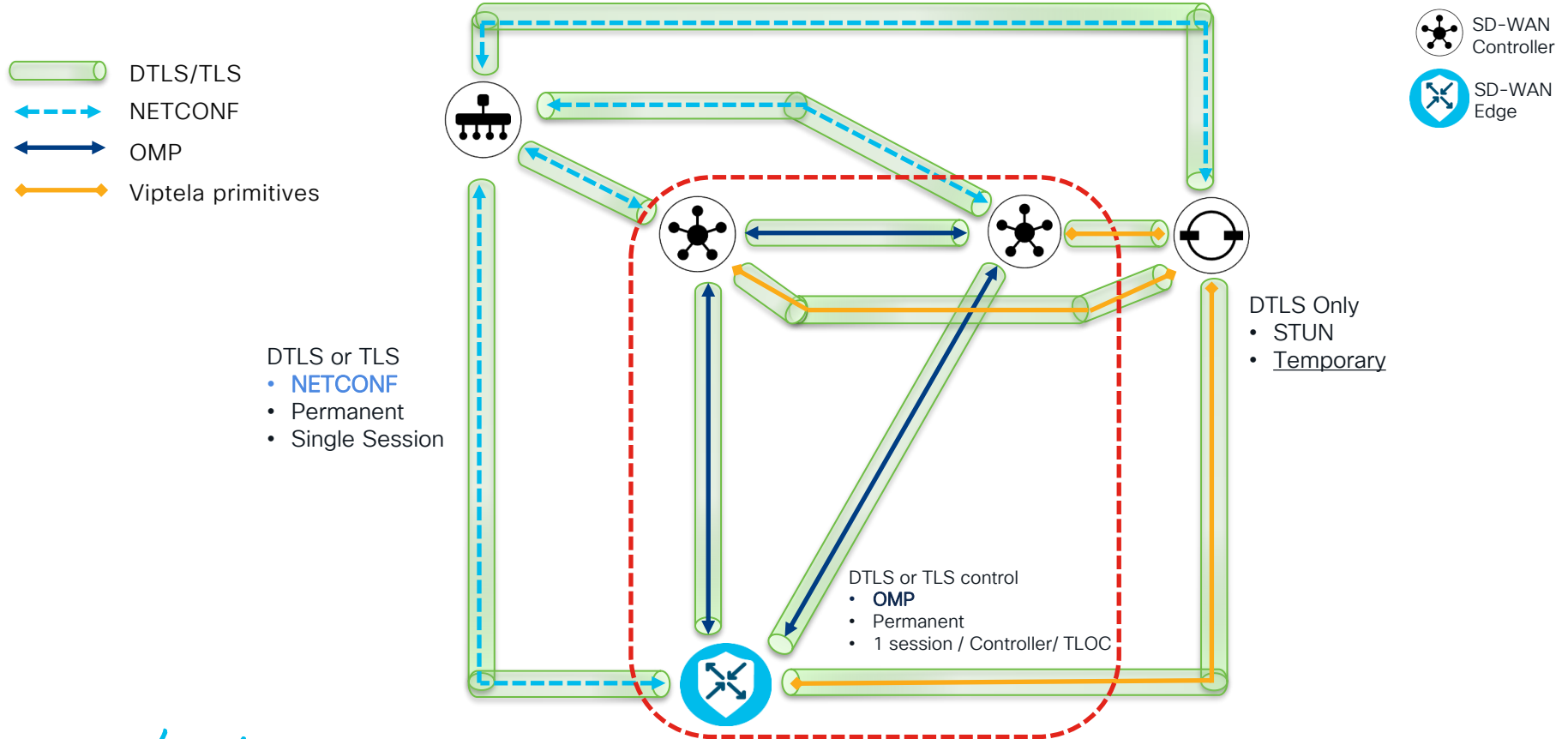
Agenda

- Part 3: SD-WAN Multicast
 - How OMP build SD-WAN Multicast tree
- Part 4: SD-WAN Path Optimization use cases
 - Full Mesh Topology
 - Hub & Spoke Topology
 - Site 100 is primary , site 200 is secondary for VPN 10 prefixes
 - Dual Hub and Spokes Topology
 - Site 100 and 200 are active together for VPN 10 prefixes

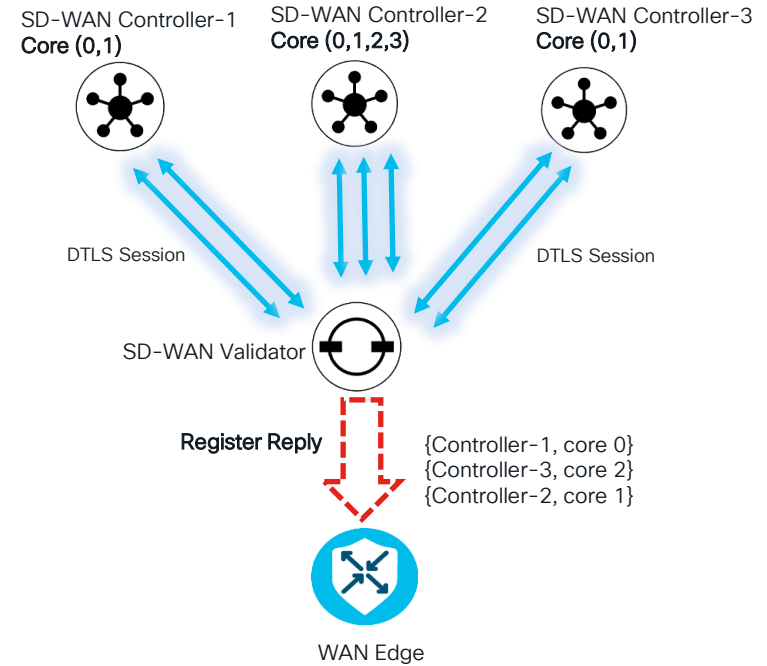
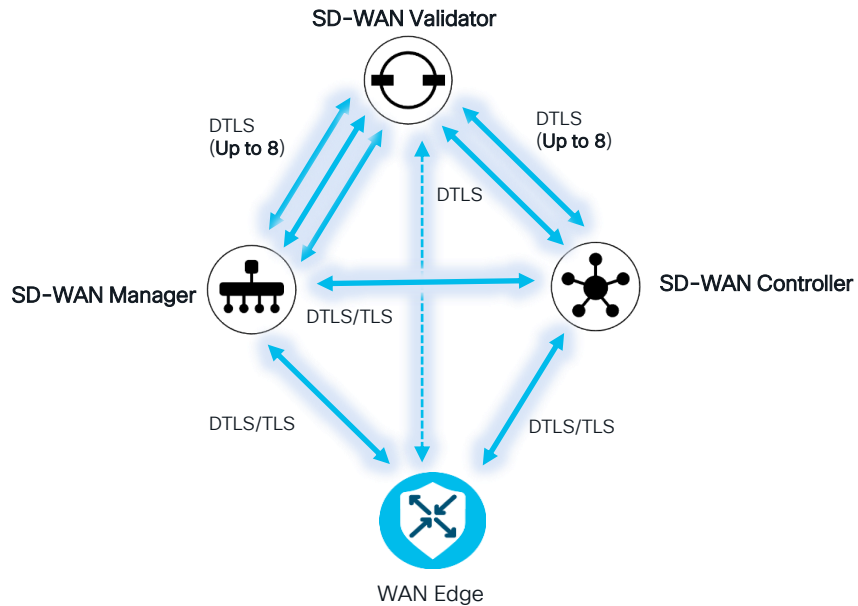
Quick SD-WAN Recap



Cisco SD-WAN Fabric overview

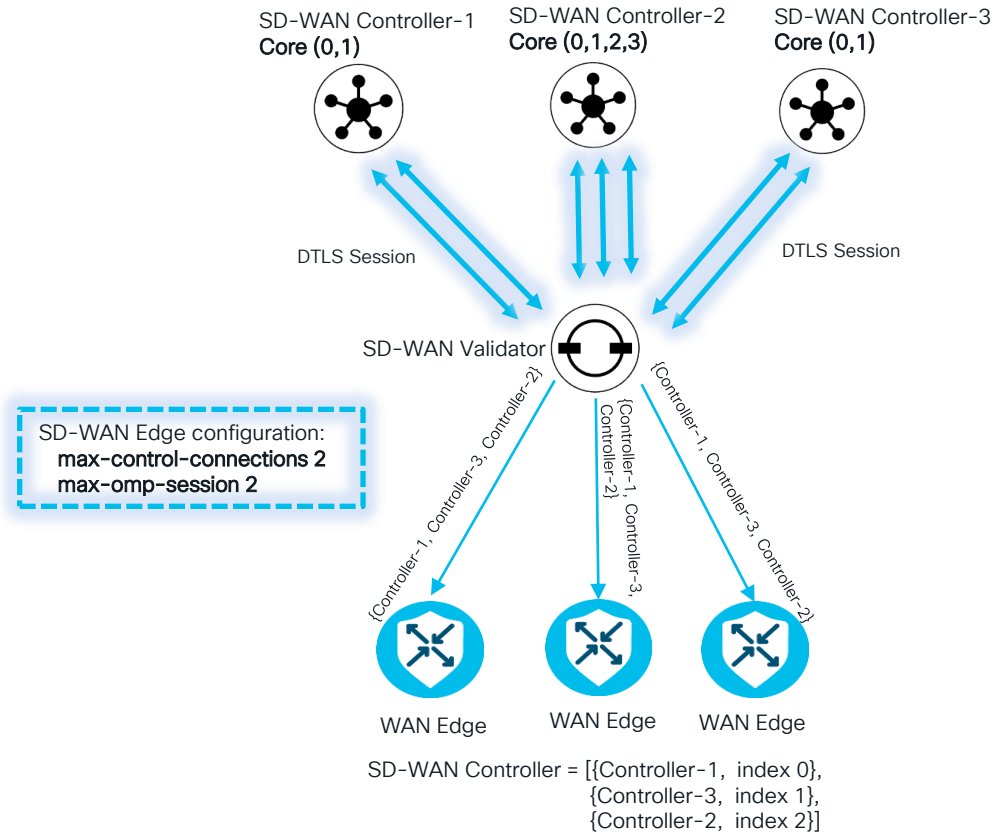


How SD-WAN Control Connections (CC) are established? 1/5

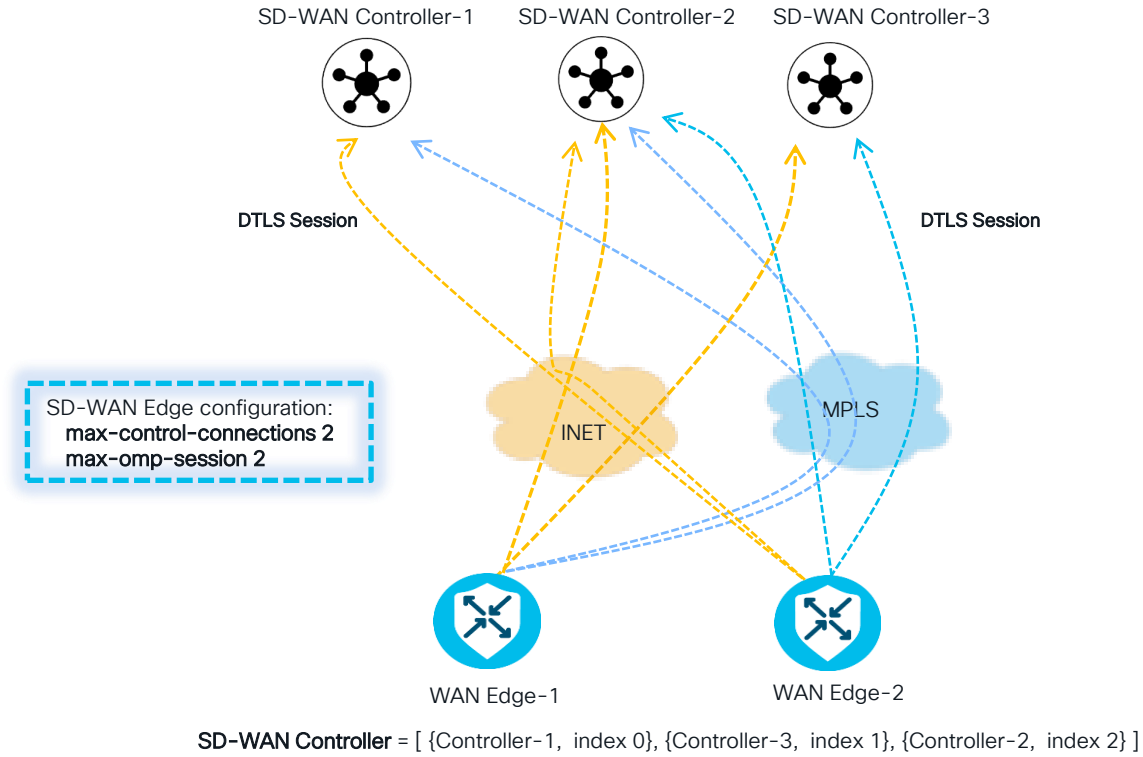


SD-WAN Controller = [{Controller-1, core 0, index 0},
{Controller-3, core 2, index 1},
{Controller-2, core 1, index 2}]

How SD-WAN Control Connections (CC) are established? 2/5



How SD-WAN Control Connections (CC) are established? 3/5



How SD-WAN Control Connections (CC) are established? 4/5

- **SD-WAN Manager** and **SD-WAN Controllers** initially contact and authenticate to SD-WAN Validator instances, forming persistent DTLS connections.
- Each **core** (up to a maximum of 8) on each **SD-WAN Manager** and **SD-WAN Controller instance** initiates and maintains a DTLS control connection to each SD-WAN Validator instance in the overlay.
- When an **WAN Edge router** initially comes up, it will attempt to establish a **transient DTLS control connection** to a **SD-WAN Validator** instance within the overlay, **over each WAN transport / TLOC**.
- As **each core of each SD-WAN Controller** instance within the overlay establishes a DTLS control connection to each SD-WAN Validator instance, **each SD-WAN Validator knows about all the SD-WAN Controller instances and cores within the overlay**.
- Within the **registration response**, the **SD-WAN Validator** will send the **list of SD-WAN Controller instances in the overlay to the individual TLOC of the SD-WAN router** and SD-WAN router will establish the control connection **from the list of SD-WAN controller until max-control-connection limit reach**.
- **Each individual TLOC** will establish the control connection with SD-WAN controllers from the list and **OMP session will be run over any of the DTLS control connections**.
- There is **no configuration** to control or **show command available** to see over which DTLS control connection the OMP session is established.
- **OMP peering is NOT bind to one DTLS connection** but rather multiple DTLS control connections to the same SD-WAN Controller instance are used, which are established over different TLOCs.
- If a given WAN transport interface on the SD-WAN Edge **router goes down**, and OMP session was running over that DTLS session, **the SD-WAN Edge router can simply switch to the other DTLS session over the other WAN transport interface/TLOC for the OMP session**.
- Once the DTLS control connections and OMP sessions are formed to the **“assigned”** SD-WAN Controller instances, each TLOC of the SD-WAN router is in **Equilibrium SD-WAN Controller Count (EVC) state**.

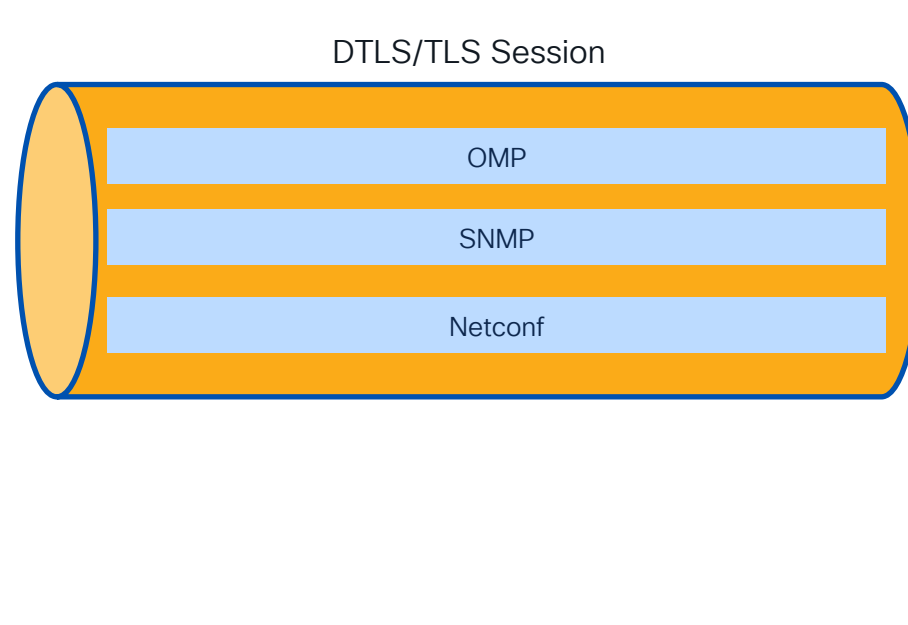
How SD-WAN Control Connections (CC) are established? 5/5



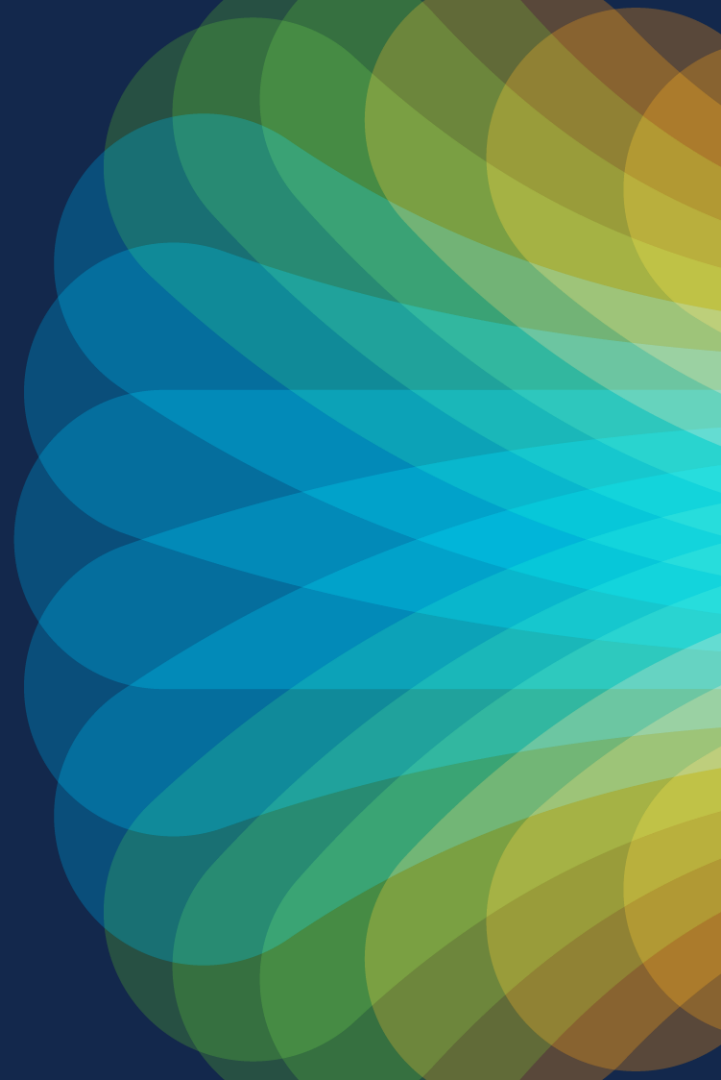
SD-WAN Edge



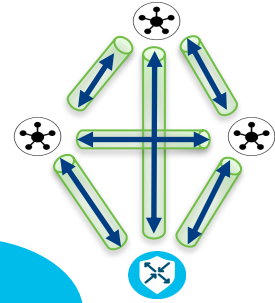
Catalyst SD-WAN Controller



Overlay Management Protocol (OMP)



Overlay Management Protocol (OMP)



OMP is a path vector routing protocol derived from BGP.

OMP peering is established inside control connection (TLS/DTLS).
TCP as transport protocol with listening on port 17900.

Leverages address families to advertise reachability.

OMP have a built-in loop avoidance capabilities

What information is propagated by OMP ?

vRoutes

Service-side
Routing info
redistributed to
Overlay
+
Attributes

TLOCs

WAN
Attachments:
system-ip
link color
encapsulation
+
Attributes

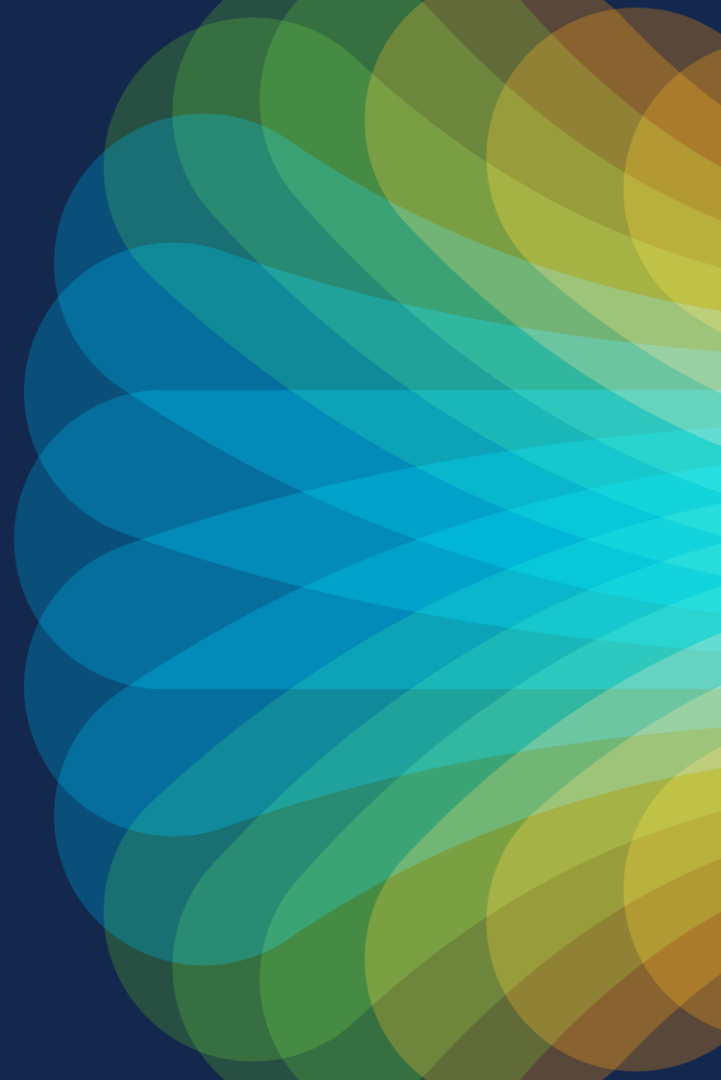
Services

Services:
Type of Service,
Location (TLOC),
Forwarding
Information

Policies

Data Policy
App-Route Policy
VPN Membership
cFlowd Template

Demystifying OMP Peering



OMP Peering Overview

- OMP peering is established **between system-IP** inside the control connection (CC).
- Only a **single peering session** is established between a WAN Edge device and a Catalyst Controller, even in the presence of multiple DTLS/TLS connections.

- Prior to initiating the exchange of OMP packets for OMP peering establishment, a **TCP connection MUST** be established between peers.
- When the WAN Edge device attempts to establish OMP peering with the Catalyst controller, the **WAN Edge operates in TCP active mode**, while the **Catalyst controller is in TCP passive mode**.

- The WAN Edge device takes the initiative in actively opening a TCP connection.
- When the Catalyst controller attempts to establish a TCP connection with another Catalyst controller, the one with the lower system-IP address operates in TCP Active mode, while the other is in TCP Passive mode.

OMP Packets

- There are **7 OMP packets** are exchanged to construct the overlay network, establish OMP peering, and disseminate routing, service, and policy information across the SDWAN fabric.
 - HANDSHAKE
 - HELLO
 - UPDATE
 - QUERY
 - ALERT
 - INFORM
 - POLICY

OMP Packet details

HANDSHAKE

Once the TCP session is established, the first packet which is exchange between peers is HANDSHAKE and it contains information like SITE-ID, HOLD-TIME and capability information like what it supports Multi-Protocol (MP) IPv4, MP IPv6, Multi-cast, services etc.. just like BGP OPEN message.

HELLO

OMP peers periodically exchange HELLO packets to indicate each peer is alive and reachable.

UPDATE

This message is used to transfer routing information between peers. OMP update message is used to advertise feasible routes that share common path attributes to a peer or to withdraw multiple unfeasible routes.

QUERY

This message is used to send a request for a specific route for which an aggregate or else specific route exists. Query message is ONLY send by the WAN Edge router once it finds out that group of prefixes received is equipped with a query attribute.

OMP Packet details

ALERT

Once the **error condition** has been deducted then peer used **ALERT** message to notify the opposite peer. The format and intention of the message is like **BGP NOTIFICATION** message.

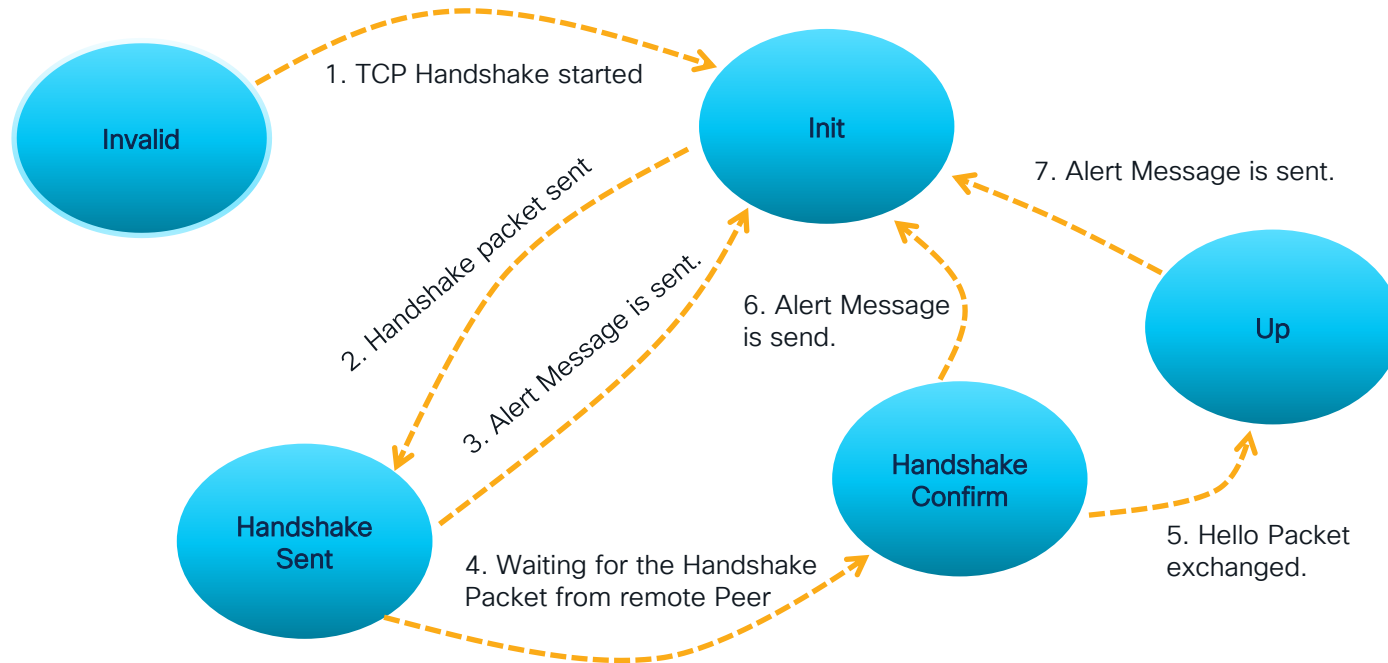
INFORM

This OMP message is associated with the **OMP GRACEFUL RESTART** feature. For example, once OMP session went down and came back up then peer send **end-of-RIB (EOR)** marker, that is an OMP INFORM packet.

POLICY

This OMP message entails all the policies including control policies, centralized data policies, Application Aware Routing (AAR), cFlow template etc...

Finite State Machine for OMP Peer Establishment



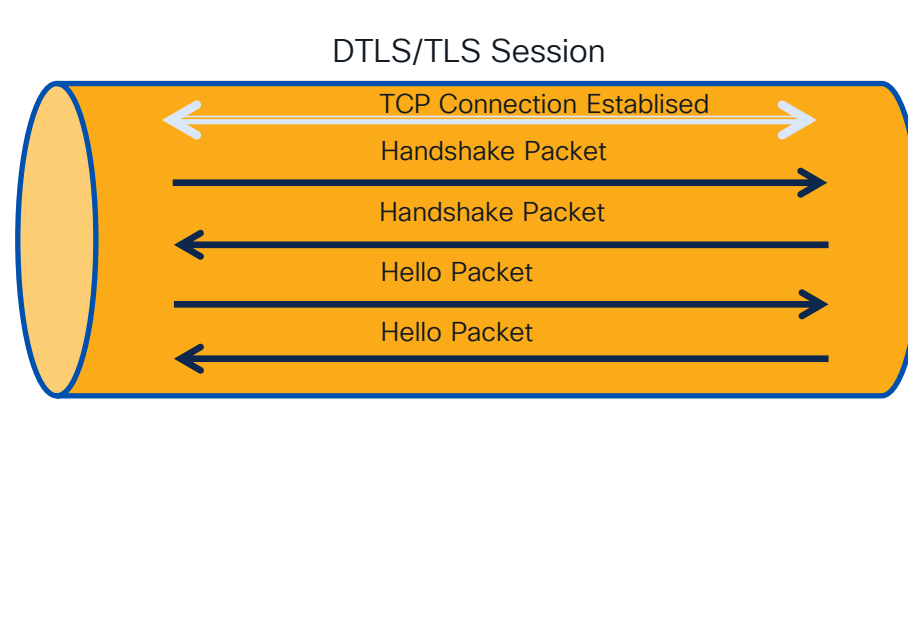
How OMP peering established?



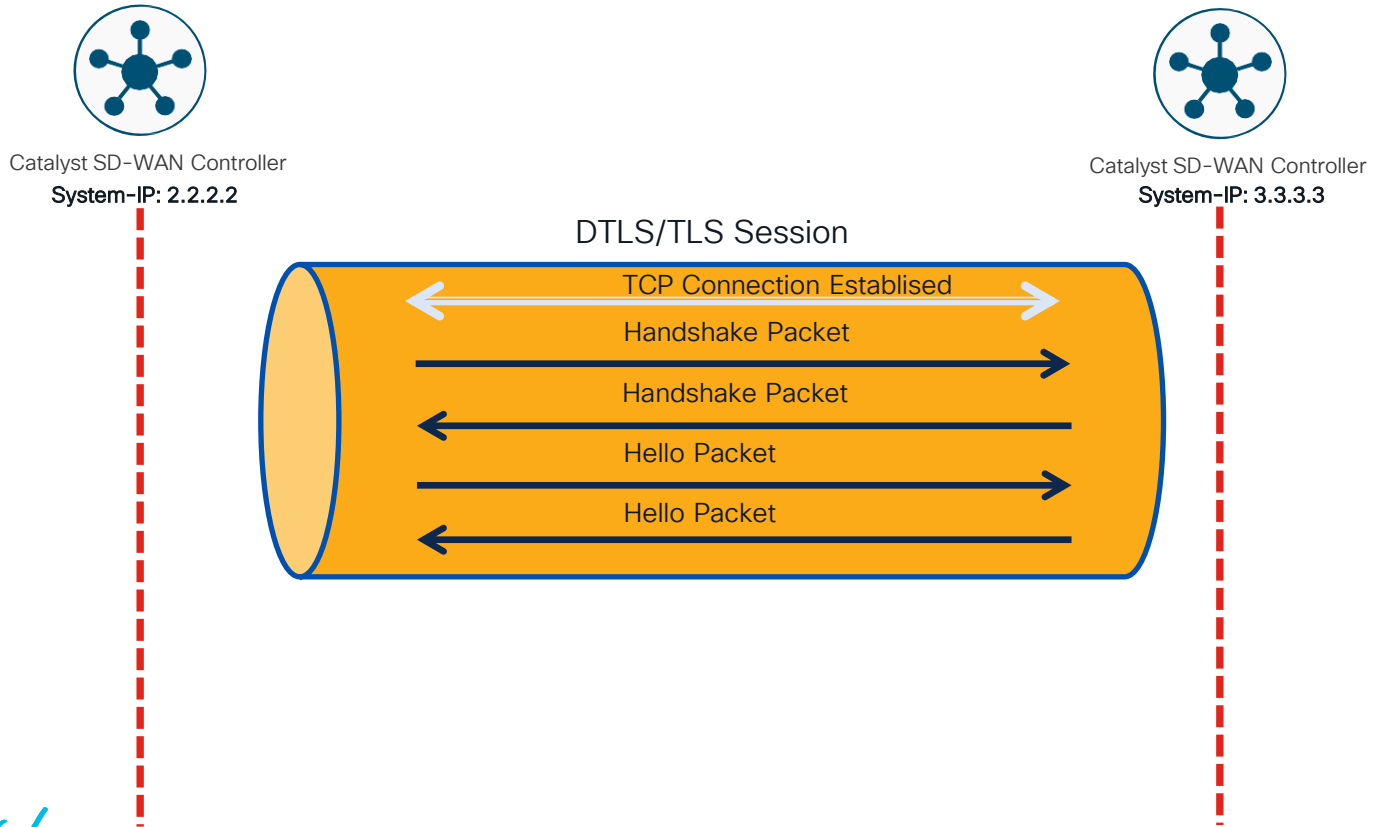
SD-WAN Edge



Catalyst SD-WAN Controller



How OMP peering established?

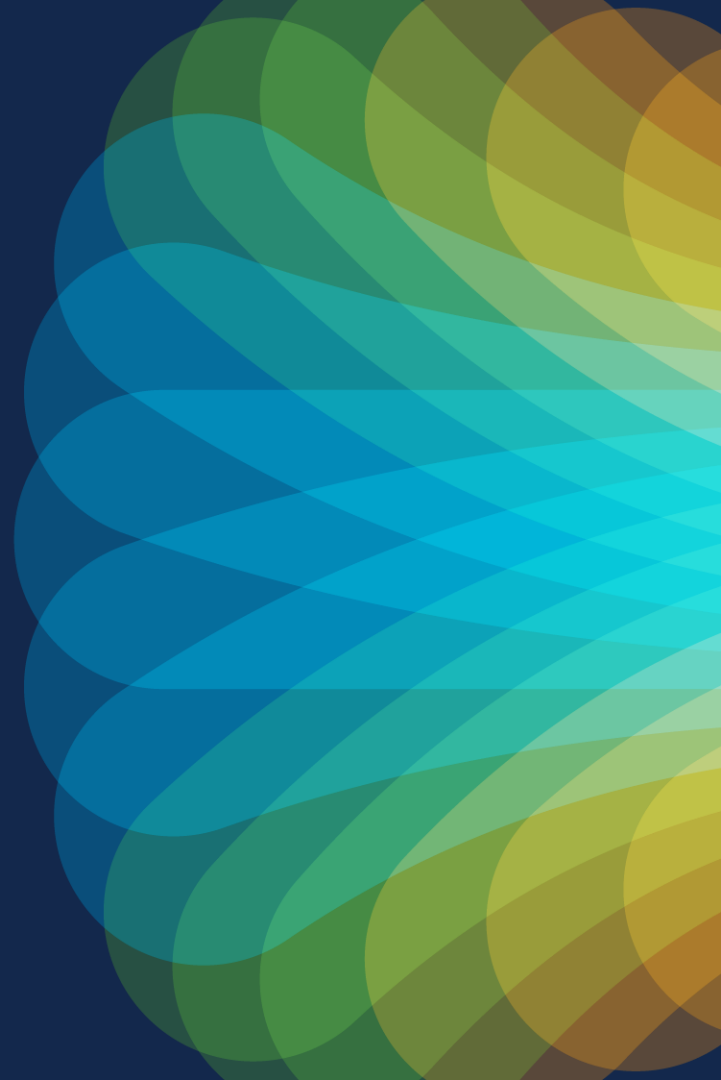


Conditions for Establishing OMP Peer Relationship



- When **Handshake packet** is received from the peer, following conditions met:
 - WAN Edge and Catalyst controllers belong to same domain (**domain-id**).
 - Atleast ONE address family matches.
 - Peer capability need to matches.
- If any of the conditions are not met, an **ALERT packet is sent** with the appropriate error code, and the OMP peering session is terminated.
- Upon receiving a **Hello packet** from a peer, the peer validates the correctness of the peer address. If there is a discrepancy, an **ALERT packet** is sent with an error code, leading to the termination of the OMP peering session.

OMP Packet Insights and commonalities with BGP



Exploring Key Similarities between BGP and OMP

BGP employs TCP as its transport protocol (**port 179**) to ensure all the transport reliability is taken care of by TCP.

BGP speakers exchange **KEEPALIVE** packets to maintain the connection keep alive.

BGP provides a mechanism to gracefully close a connection with the peer by sending a **NOTIFICATION error** message.

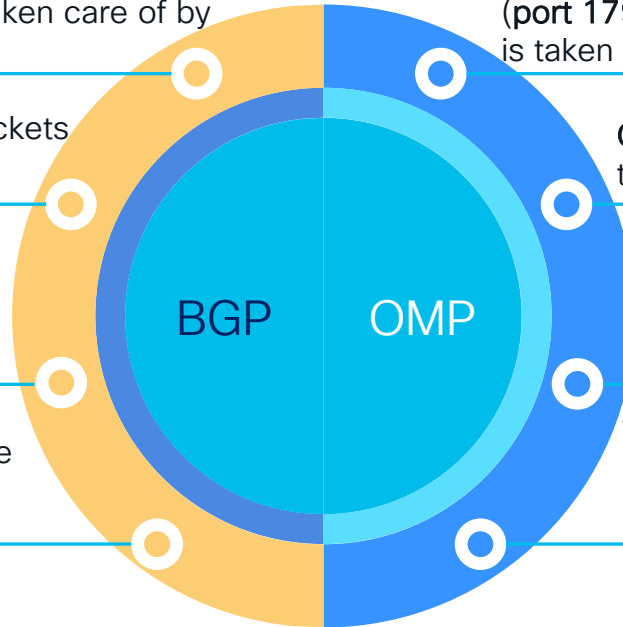
When **BGP** session has been established all candidate BGP routes are exchanged, then after that **only incremental updates** are sent.

OMP also employs TCP as its transport protocol (**port 17900**) to ensure all the transport reliability is taken care of by TCP.

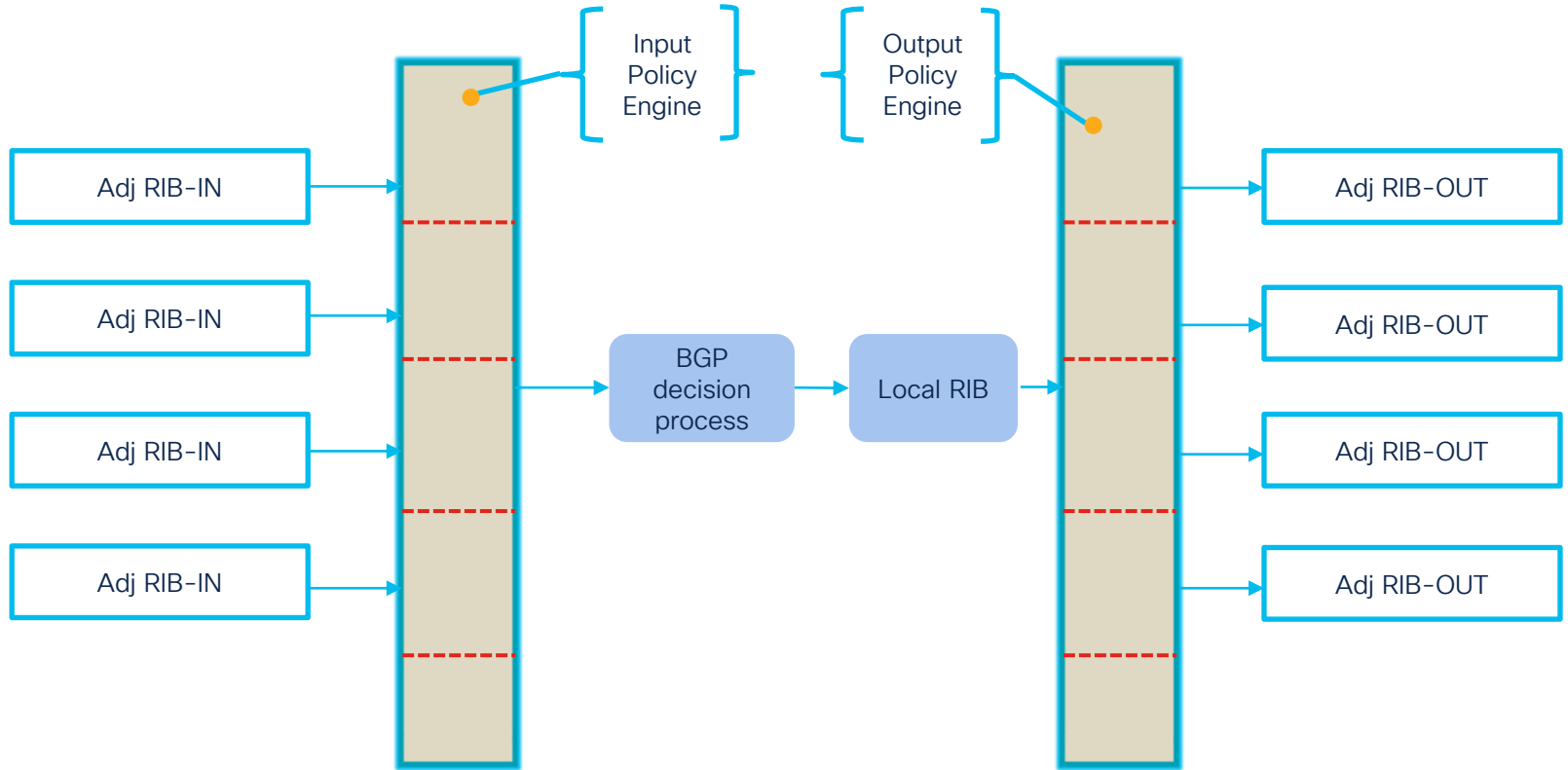
OMP peers also exchange **HELLO packets** to maintain the peering alive.

OMP also provides a mechanism to gracefully close a connection with the peer by sending a **ALERT error** message.

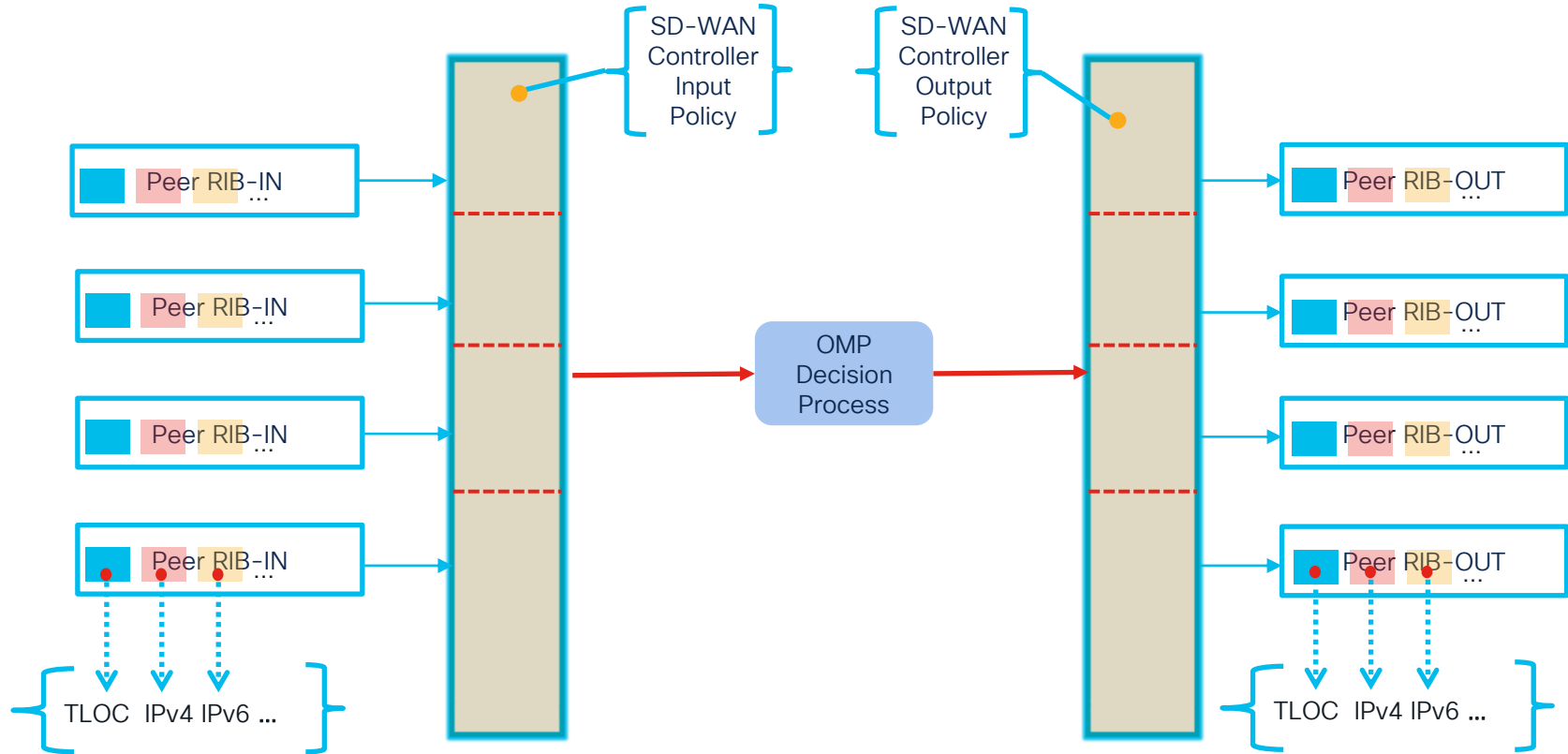
When **OMP peering** has been established and routes are exchanged then if any change **only incremental updates** are sent.



Let's recap classic BGP Routing Process



SD-WAN Controller OMP Routing Process



OMP Address Families

Address Families		Key attributes
TLOC: Transport Locators	IPv4, IPv6	WAN IP Address Color Encap IP SEC keys TLOC properties
vRoute: VPN routes	IPv4, IPV6	VPN routes Route properties
mCast: Multicast Routes	IPv4, IPv6	Multicast routes Replicators information
Service		VPN: default service Firewall
Policy		Data Policy XML
Link		BFD link properties
Cloud Express		Cloud SLA properties

Key points to remember for SD-WAN Controller OMP Routing Process



SD-WAN controller have **separate RIB-IN** and **RIB-OUT** for each OMP peer.

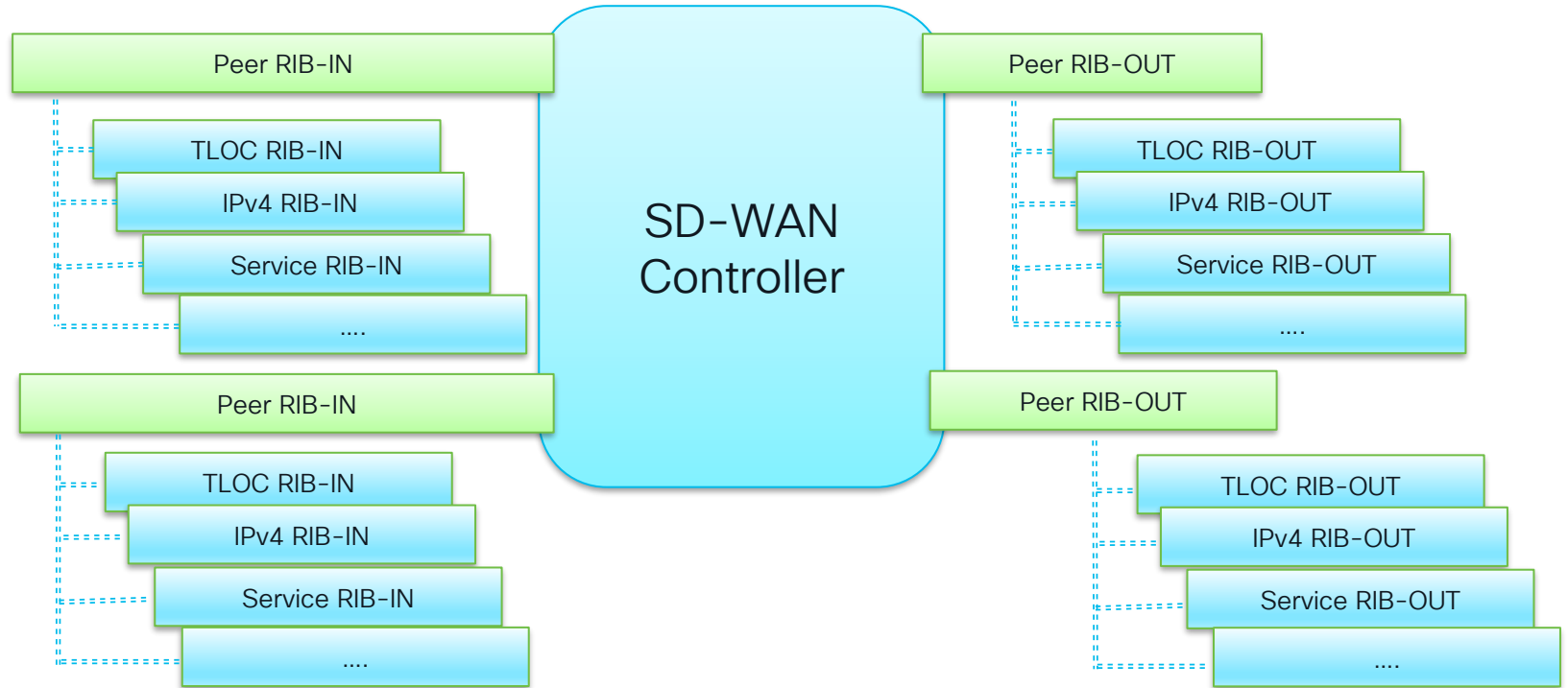
SD-WAN controller have further **separate RIB-IN and RIB-OUT** for each address family for example TLOC/prefix/service etc.. for each OMP peer.

From SD-WAN control policy perspective, when SD-WAN control policy is applied in **IN bound direction** then what ever action we perform either on TLOC/prefix/service it will be **stored and shown in RIB-IN** for that address-family for that peer and FLAG will be set accordingly.

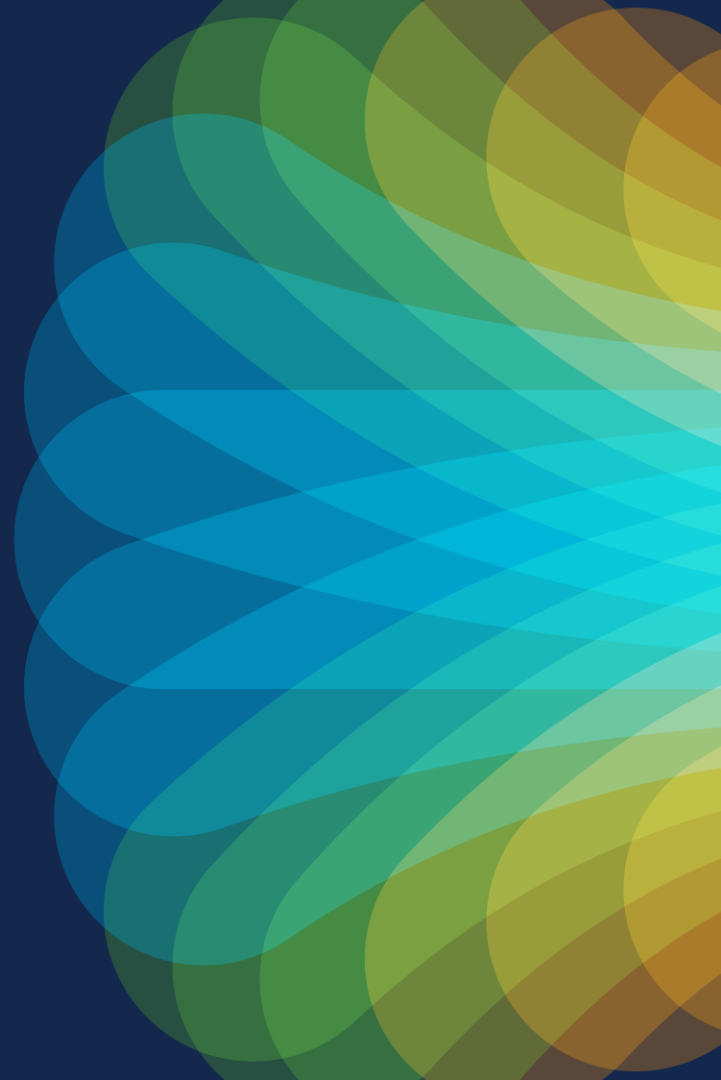
If we apply SD-WAN policy in **OUT bound direction**, then if action is **REJECT** either for TLOC/prefix/service etc.., then **RIB-OUT will NOT be generated** for that address family, so it will **NOT be advertised to other peers**.

If we apply SD-WAN control policy in **OUT bound** direction and change/modified either TLOC/prefix/service etc.. the attributes, then RIB-OUT will be generated as per our control policy/data policy and advertised to other peers.

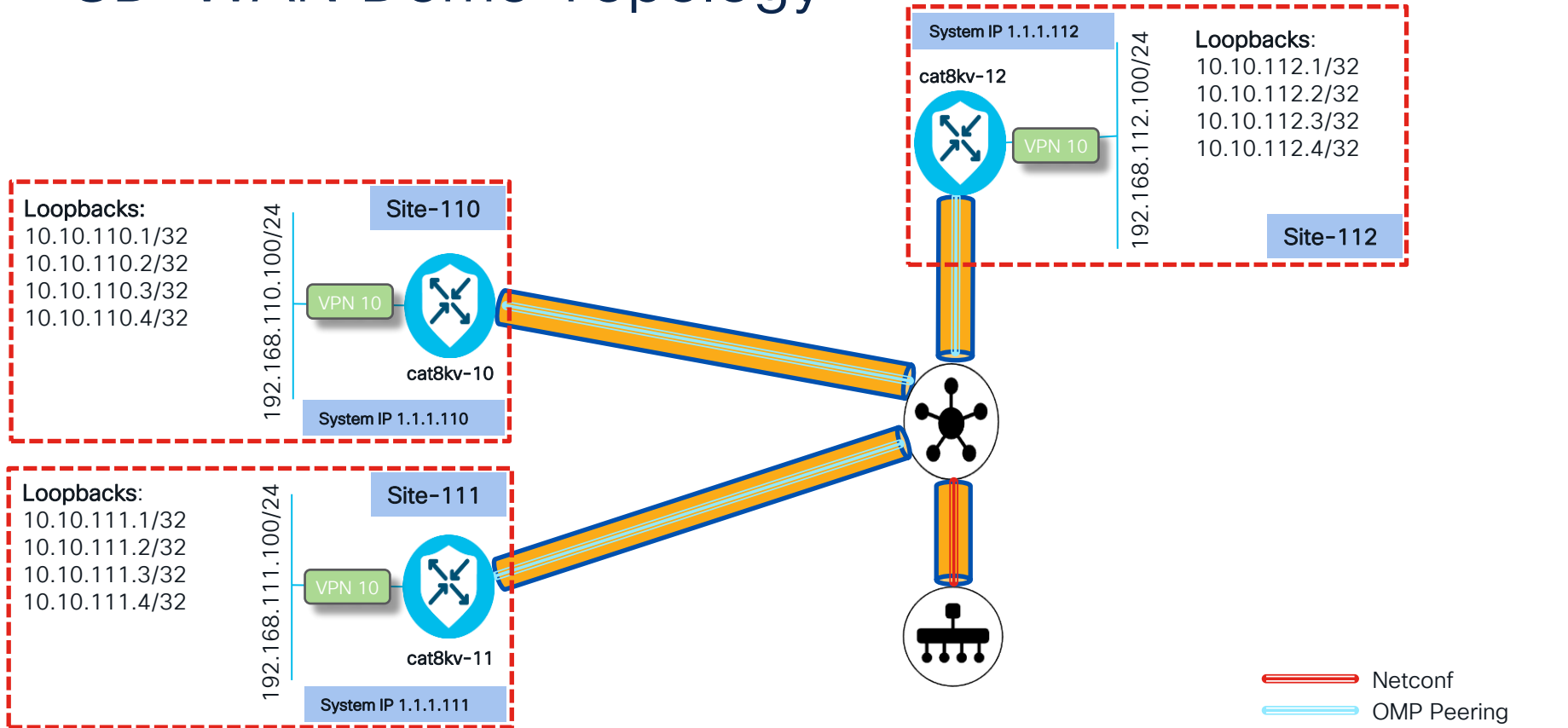
SD-WAN Controller OMP Routing Process



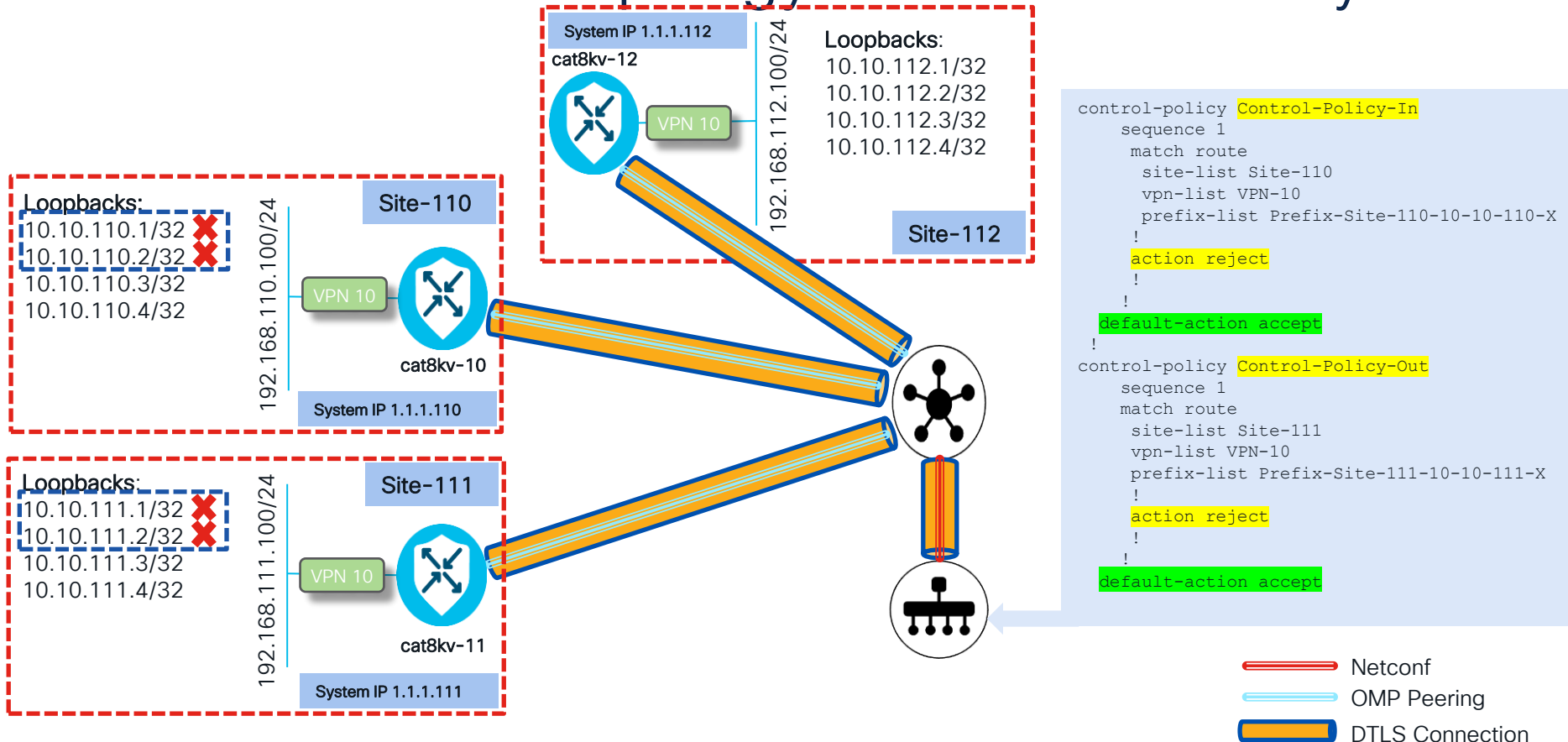
SDWAN Controller RIB-IN/RIB-OUT Demo



SD-WAN Demo Topology



SD-WAN Demo Topology with Control Policy



OMP Routes state before applying control policy

Controller-1#show omp routes vpn 10 received | begin VPN

VPN REGION	PREFIX PATH	FROM PEER	ID	LABEL	STATUS	TYPE	TLOC IP	COLOR	ENCAP	PREFERENCE	NUMBER	REGION ID	
10	10.10.110.1/32	1.1.1.110	66	1003	C,R	installed	1.1.1.110	mpls	ipsec	-	None	None	-
		1.1.1.110	81	1003	C,R	installed	1.1.1.110	private1	ipsec	-	None	None	-
		3.3.3.3	93	1003	C,R	installed	1.1.1.110	mpls	ipsec	-	None	None	-
		3.3.3.3	94	1003	C,R	installed	1.1.1.110	private1	ipsec	-	None	None	-
10	10.10.110.2/32	1.1.1.110	66	1003	C,R	installed	1.1.1.110	mpls	ipsec	-	None	None	-
		1.1.1.110	81	1003	C,R	installed	1.1.1.110	private1	ipsec	-	None	None	-
		3.3.3.3	89	1003	C,R	installed	1.1.1.110	mpls	ipsec	-	None	None	-
		3.3.3.3	94	1003	C,R	installed	1.1.1.110	private1	ipsec	-	None	None	-
10	10.10.110.3/32	1.1.1.110	66	1003	C,R	installed	1.1.1.110	mpls	ipsec	-	None	None	-
		1.1.1.110	81	1003	C,R	installed	1.1.1.110	private1	ipsec	-	None	None	-
		3.3.3.3	48	1003	C,R	installed	1.1.1.110	mpls	ipsec	-	None	None	-
		3.3.3.3	55	1003	C,R	installed	1.1.1.110	private1	ipsec	-	None	None	-
10	10.10.110.4/32	1.1.1.110	66	1003	C,R	installed	1.1.1.110	mpls	ipsec	-	None	None	-
		1.1.1.110	81	1003	C,R	installed	1.1.1.110	private1	ipsec	-	None	None	-
		3.3.3.3	48	1003	C,R	installed	1.1.1.110	mpls	ipsec	-	None	None	-
		3.3.3.3	55	1003	C,R	installed	1.1.1.110	private1	ipsec	-	None	None	-
10	10.10.111.1/32	1.1.1.111	66	1003	C,R	installed	1.1.1.111	mpls	ipsec	-	None	None	-
		1.1.1.111	81	1003	C,R	installed	1.1.1.111	private1	ipsec	-	None	None	-
		3.3.3.3	15	1003	C,R	installed	1.1.1.111	mpls	ipsec	-	None	None	-
		3.3.3.3	16	1003	C,R	installed	1.1.1.111	private1	ipsec	-	None	None	-
10	10.10.111.2/32	1.1.1.111	66	1003	C,R	installed	1.1.1.111	mpls	ipsec	-	None	None	-
		1.1.1.111	81	1003	C,R	installed	1.1.1.111	private1	ipsec	-	None	None	-
		3.3.3.3	15	1003	C,R	installed	1.1.1.111	mpls	ipsec	-	None	None	-
		3.3.3.3	16	1003	C,R	installed	1.1.1.111	private1	ipsec	-	None	None	-
10	10.10.111.3/32	1.1.1.111	66	1003	C,R	installed	1.1.1.111	mpls	ipsec	-	None	None	-
		1.1.1.111	81	1003	C,R	installed	1.1.1.111	private1	ipsec	-	None	None	-
		3.3.3.3	15	1003	C,R	installed	1.1.1.111	mpls	ipsec	-	None	None	-
		3.3.3.3	16	1003	C,R	installed	1.1.1.111	private1	ipsec	-	None	None	-
10	10.10.111.4/32	1.1.1.111	66	1003	C,R	installed	1.1.1.111	mpls	ipsec	-	None	None	-
		1.1.1.111	81	1003	C,R	installed	1.1.1.111	private1	ipsec	-	None	None	-
		3.3.3.3	15	1003	C,R	installed	1.1.1.111	mpls	ipsec	-	None	None	-
		3.3.3.3	16	1003	C,R	installed	1.1.1.111	private1	ipsec	-	None	None	-

OMP Routes state after applying Inbound control policy

Controller-1#show omp routes vpn 10 received | begin VPN

VPN REGION	PREFIX PATH	FROM PEER	ID	LABEL	STATUS	TYPE	TLOC IP	COLOR	ENCAP	PREFERENCE	NUMBER	REGION ID
10	10.10.110.1/32	1.1.1.110	66	1003	Rej,R,Inv	installed	1.1.1.110	mpls	ipsec	-	None	None
		1.1.1.110	81	1003	Rej,R,Inv	installed	1.1.1.110	private1	ipsec	-	None	None
10	10.10.110.2/32	1.1.1.110	66	1003	Rej,R,Inv	installed	1.1.1.110	mpls	ipsec	-	None	None
		1.1.1.110	81	1003	Rej,R,Inv	installed	1.1.1.110	private1	ipsec	-	None	None
10	10.10.110.3/32	1.1.1.110	66	1003	C,R	installed	1.1.1.110	mpls	ipsec	-	None	None
		1.1.1.110	81	1003	C,R	installed	1.1.1.110	private1	ipsec	-	None	None
		3.3.3.3	48	1003	C,R	installed	1.1.1.110	mpls	ipsec	-	None	None
		3.3.3.3	55	1003	C,R	installed	1.1.1.110	private1	ipsec	-	None	None
10	10.10.110.4/32	1.1.1.110	66	1003	C,R	installed	1.1.1.110	mpls	ipsec	-	None	None
		1.1.1.110	81	1003	C,R	installed	1.1.1.110	private1	ipsec	-	None	None
		3.3.3.3	48	1003	C,R	installed	1.1.1.110	mpls	ipsec	-	None	None
		3.3.3.3	55	1003	C,R	installed	1.1.1.110	private1	ipsec	-	None	None
10	10.10.111.1/32	1.1.1.111	66	1003	C,R	installed	1.1.1.111	mpls	ipsec	-	None	None
		1.1.1.111	81	1003	C,R	installed	1.1.1.111	private1	ipsec	-	None	None
		3.3.3.3	15	1003	C,R	installed	1.1.1.111	mpls	ipsec	-	None	None
		3.3.3.3	16	1003	C,R	installed	1.1.1.111	private1	ipsec	-	None	None
10	10.10.111.2/32	1.1.1.111	66	1003	C,R	installed	1.1.1.111	mpls	ipsec	-	None	None
		1.1.1.111	81	1003	C,R	installed	1.1.1.111	private1	ipsec	-	None	None
		3.3.3.3	15	1003	C,R	installed	1.1.1.111	mpls	ipsec	-	None	None
		3.3.3.3	16	1003	C,R	installed	1.1.1.111	private1	ipsec	-	None	None
10	10.10.111.3/32	1.1.1.111	66	1003	C,R	installed	1.1.1.111	mpls	ipsec	-	None	None
		1.1.1.111	81	1003	C,R	installed	1.1.1.111	private1	ipsec	-	None	None
		3.3.3.3	15	1003	C,R	installed	1.1.1.111	mpls	ipsec	-	None	None
		3.3.3.3	16	1003	C,R	installed	1.1.1.111	private1	ipsec	-	None	None
10	10.10.111.4/32	1.1.1.111	66	1003	C,R	installed	1.1.1.111	mpls	ipsec	-	None	None
		1.1.1.111	81	1003	C,R	installed	1.1.1.111	private1	ipsec	-	None	None
		3.3.3.3	15	1003	C,R	installed	1.1.1.111	mpls	ipsec	-	None	None
		3.3.3.3	16	1003	C,R	installed	1.1.1.111	private1	ipsec	-	None	None

OMP Routes state after applying Inbound control policy

```
Controller-1# show omp routes vpn 10 advertised | begin VPN
```

VPN	PREFIX	TO PEER
10	10.10.110.3/32	1.1.1.111 1.1.1.112 1.1.1.113 1.1.1.114 1.1.1.115 1.1.1.116 3.3.3.3
10	10.10.110.4/32	1.1.1.111 1.1.1.112 1.1.1.113 1.1.1.114 1.1.1.115 1.1.1.116 3.3.3.3
10	10.10.111.1/32	1.1.1.110 1.1.1.113 1.1.1.114 1.1.1.115 1.1.1.116 3.3.3.3
10	10.10.111.2/32	1.1.1.110 1.1.1.113 1.1.1.114 1.1.1.115 1.1.1.116 3.3.3.3
10	10.10.111.3/32	1.1.1.110 1.1.1.112 1.1.1.113 1.1.1.114 1.1.1.115 1.1.1.116 3.3.3.3
10	10.10.111.4/32	1.1.1.110 1.1.1.112 1.1.1.113 1.1.1.114 1.1.1.115 1.1.1.116 3.3.3.3

```
Controller-1# show support omp rib vroute 10:10.10.110.1/32
```

```
Looking up vroute 10.10.110.1/32 in 10
```

```
RIB-Entry: (0x7f131609c780) ROUTE-IPV4 Flags: (0x0) , rcv-attr-count 2, adv-attr-count 0  
ri-peer-tree: 0x7f131609c7d8(2), ro-peer-tree: 0x7f131609c800(0), ro-ri-id-tree: 0x7f131609c838(0),  
Scheduled: 72, Version: 1195, ro-cache-ri-id-tree: 0x7f13160a2090(0), ro-cache-policy-name-tree:  
0x7f13160a20c0(0) VPN-ID: 10, Prefix: 10.10.110.1/32
```

```
Region-Info Region-ID: 65534
```

```
RIB-IN: (0x7f13153631e0, prev: (nil), next: 0x7f131532e380), Peer: 1.1.1.110, ID: 355, updated: Mon Nov 20  
12:41:32 2023
```

```
Path-id: 66, Label: 1003 Affinity Number: 0 TLOC-pref: 0 TLOC-stale: 0 version: 1 (stale: 0) Management-  
Region: False
```

```
Lost-to-peer: ::, Lost-to-path-id: 0, Loss-Reason: None(0)
```

```
Rcv-Attr: 0x7f1313401500, Flags: (0x28) REJECT RESOLVED
```

```
Attribute: (0x7f1313401500), ROUTE-IPV4, Length: 1184, Ref: 2
```

```
Flags: (0x8000c25) WEIGHT TLOC SITE-ID OVERLAY-ID ORIGIN ORIGINATOR
```

```
Region-id: 65534, Secondary-Region-id: 65535, Orig-Access-Region-id: 65534, Sub-Region-ID: 0, Pref: 0,
```

```
Weight: 1, Tag: 0, Stale: 0 Version: 0, Restrict: 0, on-Demand: 0, Domain: 0, BR-Preference: 0, Affinity-  
Group-Number:0, MRF-Route-Originator:None , Derived Affinity-group-number: 0 Distance: 0, Site-ID: 110,
```

```
Carrier: 0, Query: 0, Gen-ID: 0x0, Border: 0 Overlay: 1 Site-Type: 0 0 0 0
```

```
Originator: 1.1.1.110
```

```
Origin: Protocol: ospf[4], Sub-Type: intra-area[1], Metric: 1
```

```
TLOC: (0x7f1316366500) 1.1.1.110 : mpls : ipsec
```

```
RIB-IN: (0x7f131532e380, prev: 0x7f13153631e0, next: (nil)), Peer: 1.1.1.110, ID: 371, updated: Mon Nov 20  
12:41:32 2023
```

```
Path-id: 81, Label: 1003 Affinity Number: 0 TLOC-pref: 0 TLOC-stale: 0 version: 1 (stale: 0) Management-  
Region: False
```

```
Lost-to-peer: 1.1.1.110, Lost-to-path-id: 66, Loss-Reason: TLOC ID(10)
```

```
Rcv-Attr: 0x7f1313453c00, Flags: (0x28) REJECT RESOLVED
```

```
Attribute: (0x7f1313453c00), ROUTE-IPV4, Length: 1184, Ref: 2
```

```
Flags: (0x8000c25) WEIGHT TLOC SITE-ID OVERLAY-ID ORIGIN ORIGINATOR
```

```
Region-id: 65534, Secondary-Region-id: 65535, Orig-Access-Region-id: 65534, Sub-Region-ID: 0, Pref: 0,
```

```
Weight: 1, Tag: 0, Stale: 0 Version: 0, Restrict: 0, on-Demand: 0, Domain: 0, BR-Preference: 0, Affinity-  
Group-Number:0, MRF-Route-Originator:None , Derived Affinity-group-number: 0 Distance: 0, Site-ID: 110,
```

```
Carrier: 0, Query: 0, Gen-ID: 0x0, Border: 0 Overlay: 1 Site-Type: 0 0 0 0
```

```
Originator: 1.1.1.110
```

```
Origin: Protocol: ospf[4], Sub-Type: intra-area[1], Metric: 1
```

```
TLOC: (0x7f13163665a0) 1.1.1.110 : private1 : ipsec
```

OMP Routes state after applying Inbound control policy

```
Controller-1# show omp routes vpn 10 advertised | begin VPN
VPN      PREFIX          TO PEER
-----
10       10.10.110.3/32  1.1.1.111
          1.1.1.112
          1.1.1.113
          1.1.1.114
          1.1.1.115
          1.1.1.116
          3.3.3.3
10       10.10.110.4/32  1.1.1.111
          1.1.1.112
          1.1.1.113
          1.1.1.114
          1.1.1.115
          1.1.1.116
          3.3.3.3
10       10.10.111.1/32  1.1.1.110
          1.1.1.113
          1.1.1.114
          1.1.1.115
          1.1.1.116
          3.3.3.3
10       10.10.111.2/32  1.1.1.110
          1.1.1.113
          1.1.1.114
          1.1.1.115
          1.1.1.116
          3.3.3.3
10       10.10.111.3/32  1.1.1.110
          1.1.1.112
          1.1.1.113
          1.1.1.114
          1.1.1.115
          1.1.1.116
          3.3.3.3
10       10.10.111.4/32  1.1.1.110
          1.1.1.112
          1.1.1.113
          1.1.1.114
          1.1.1.115
          1.1.1.116
          3.3.3.3
```

OMP Routes state after applying Inbound control policy

```
Controller-1# show support omp rib vroute 10:10.10.110.3/32
```

```
Looking up vroute 10.10.110.3/32 in 10
```

```
RIB-IN: (0x7f1315363020, prev: (nil), next: 0x7f13153013a0), Peer: 1.1.1.110, ID: 357, updated: Mon Nov 20 12:41:32 2023
```

```
Path-id: 66, Label: 1003 Affinity Number: 0 TLOC-pref: 0 TLOC-stale: 0 version: 9 (stale: 0) Management-Region: False
```

```
Lost-to-peer: ::, Lost-to-path-id: 0, Loss-Reason: None(0)
```

```
Rcv-Attr: 0x7f1316b16e00, Flags: (0x21) CHOSEN RESOLVED
```

```
Attribute: (0x7f1316b16e00), ROUTE-IPV4, Length: 1184, Ref: 10
```

```
Flags: (0x8000c25) WEIGHT TLOC SITE-ID OVERLAY-ID ORIGIN ORIGINATOR
```

```
Region-id: 65534, Secondary-Region-id: 65535, Orig-Access-Region-id: 65534, Sub-Region-ID: 0, Pref: 0, Weight: 1, Tag: 0, Stale: 0 Version: 0, Restrict: 0, on-Demand: 0, Domain: 0, BR-Preference: 0, Affinity-Group-Number:0, MRF-Route-Originator:None , Derived Affinity-group-number: 0 Distance: 0, Site-ID: 110,
```

```
Carrier: 0, Query: 0, Gen-ID: 0x0, Border: 0 Overlay: 1 Site-Type: 0 0 0 0
```

```
Originator: 1.1.1.110
```

```
Origin: Protocol: ospf[4], Sub-Type: intra-area[1], Metric: 11
```

```
TLOC: (0x7f1316366500) 1.1.1.110 : mpls : ipsec
```

```
RIB-IN: (0x7f13153013a0, prev: 0x7f1315363020, next: 0x7f131635bba0), Peer: 1.1.1.110, ID: 373, updated: Mon Nov 20 12:41:32 2023
```

```
Path-id: 81, Label: 1003 Affinity Number: 0 TLOC-pref: 0 TLOC-stale: 0 version: 9 (stale: 0) Management-Region: False
```

```
Lost-to-peer: 1.1.1.110, Lost-to-path-id: 66, Loss-Reason: TLOC ID(10)
```

```
Rcv-Attr: 0x7f1313405b00, Flags: (0x21) CHOSEN RESOLVED
```

```
Attribute: (0x7f1313405b00), ROUTE-IPV4, Length: 1184, Ref: 10
```

```
Flags: (0x8000c25) WEIGHT TLOC SITE-ID OVERLAY-ID ORIGIN ORIGINATOR
```

```
Region-id: 65534, Secondary-Region-id: 65535, Orig-Access-Region-id: 65534, Sub-Region-ID: 0, Pref: 0, Weight: 1, Tag: 0, Stale: 0 Version: 0, Restrict: 0, on-Demand: 0, Domain: 0, BR-Preference: 0, Affinity-Group-Number:0, MRF-Route-Originator:None , Derived Affinity-group-number: 0 Distance: 0, Site-ID: 110,
```

```
Carrier: 0, Query: 0, Gen-ID: 0x0, Border: 0 Overlay: 1 Site-Type: 0 0 0 0
```

```
Originator: 1.1.1.110
```

```
Origin: Protocol: ospf[4], Sub-Type: intra-area[1], Metric: 11
```

```
TLOC: (0x7f13163665a0) 1.1.1.110 : private1 : ipsec
```

```
RIB-IN: (0x7f131635bba0, prev: 0x7f13153013a0, next: 0x7f1316359b40), Peer: 3.3.3.3, ID: 361, updated: Tue Nov 14 16:57:03 2023
```

```
Path-id: 48, Label: 1003 Affinity Number: 0 TLOC-pref: 0 TLOC-stale: 0 version: 9 (stale: 0) Management-Region: False
```

```
Lost-to-peer: 1.1.1.110, Lost-to-path-id: 81, Loss-Reason: Personality(2)
```

```
Rcv-Attr: 0x7f1316b16e00, Flags: (0x21) CHOSEN RESOLVED
```

```
Attribute: (0x7f1316b16e00), ROUTE-IPV4, Length: 1184, Ref: 10
```

```
Flags: (0x8000c25) WEIGHT TLOC SITE-ID OVERLAY-ID ORIGIN ORIGINATOR
```

```
Region-id: 65534, Secondary-Region-id: 65535, Orig-Access-Region-id: 65534, Sub-Region-ID: 0, Pref: 0, Weight: 1, Tag: 0, Stale: 0 Version: 0, Restrict: 0, on-Demand: 0, Domain: 0, BR-Preference: 0, Affinity-Group-Number:0, MRF-Route-Originator:None , Derived Affinity-group-number: 0 Distance: 0, Site-ID: 110,
```

```
Carrier: 0, Query: 0, Gen-ID: 0x0, Border: 0 Overlay: 1 Site-Type: 0 0 0 0
```

```
Originator: 1.1.1.110
```

```
Origin: Protocol: ospf[4], Sub-Type: intra-area[1], Metric: 11
```

```
TLOC: (0x7f1316366500) 1.1.1.110 : mpls : ipsec
```

```
<continue in next slide>
```

OMP Routes state after applying Inbound control policy

```
Controller-1# show support omp rib vroute 10.10.10.110.3/32
```

```
...
RIB-IN: (0x7f1316359b40, prev: 0x7f131635bba0, next: (nil)), Peer: 3.3.3.3, ID: 386, updated: Tue Nov 14 16:58:09 2023
  Path-id: 55, Label: 1003 Affinity Number: 0 TLOC-pref: 0 TLOC-stale: 0 version: 9 (stale: 0) Management-Region: False
  Lost-to-peer: 1.1.1.110, Lost-to-path-id: 81, Loss-Reason: Personality(2)
  Rcv-Attr: 0x7f1313405b00, Flags: (0x21) CHOSEN RESOLVED
  Attribute: (0x7f1313405b00), ROUTE-IPV4, Length: 1184, Ref: 10
  Flags: (0x8000c25) WEIGHT TLOC SITE-ID OVERLAY-ID ORIGIN ORIGINATOR
  Region-id: 65534, Secondary-Region-id: 65535, Orig-Access-Region-id: 65534, Sub-Region-ID: 0, Pref: 0, Weight: 1, Tag: 0, Stale: 0 Version: 0, Restrict: 0,
on-Demand: 0, Domain: 0, BR-Preference: 0, Affinity-Group-Number:0, MRF-Route-Originator:None , Derived Affinity-group-number: 0 Distance: 0, Site-ID: 110,
Carrier: 0, Query: 0, Gen-ID: 0x0, Border: 0 Overlay: 1 Site-Type: 0 0 0 0
  Originator: 1.1.1.110
  Origin: Protocol: ospf[4], Sub-Type: intra-area[1], Metric: 11
  TLOC: (0x7f13163665a0) 1.1.1.110 : privatel : ipsec
RIB-OUT: (0x7f13192b0080), RI-ID: 357, Peer: 1.1.1.111 Path-id: 48, Label: 1003, Flags: (0x1) ADV Common (0x7f1313406680) Common Ref: 7
  Attribute: (0x7f13134a9200), ROUTE-IPV4, Length: 1184, Ref: 6
  Flags: (0x8000c25) WEIGHT TLOC SITE-ID OVERLAY-ID ORIGIN ORIGINATOR
  Region-id: 65534, Secondary-Region-id: 65535, Orig-Access-Region-id: 65534, Sub-Region-ID: 0, Pref: 0, Weight: 1, Tag: 0, Stale: 0 Version: 0, Restrict: 0,
on-Demand: 0, Domain: 0, BR-Preference: 0, Affinity-Group-Number:0, MRF-Route-Originator:None , Derived Affinity-group-number: 0 Distance: 0, Site-ID: 110,
Carrier: 0, Query: 0, Gen-ID: 0x0, Border: 0 Overlay: 1 Site-Type: 0 0 0 0
  Originator: 1.1.1.110
  Origin: Protocol: ospf[4], Sub-Type: intra-area[1], Metric: 11
  TLOC: ((nil)) 1.1.1.110 : mpls : ipsec
RIB-OUT: (0x7f13192a8fa0), RI-ID: 373, Peer: 1.1.1.111 Path-id: 55, Label: 1003, Flags: (0x1) ADV Common (0x7f1316ac5040) Common Ref: 7
  Attribute: (0x7f1317148e00), ROUTE-IPV4, Length: 1184, Ref: 6
  Flags: (0x8000c25) WEIGHT TLOC SITE-ID OVERLAY-ID ORIGIN ORIGINATOR
  Region-id: 65534, Secondary-Region-id: 65535, Orig-Access-Region-id: 65534, Sub-Region-ID: 0, Pref: 0, Weight: 1, Tag: 0, Stale: 0 Version: 0, Restrict: 0,
on-Demand: 0, Domain: 0, BR-Preference: 0, Affinity-Group-Number:0, MRF-Route-Originator:None , Derived Affinity-group-number: 0 Distance: 0, Site-ID: 110,
Carrier: 0, Query: 0, Gen-ID: 0x0, Border: 0 Overlay: 1 Site-Type: 0 0 0 0
  Originator: 1.1.1.110
  Origin: Protocol: ospf[4], Sub-Type: intra-area[1], Metric: 11
  TLOC: ((nil)) 1.1.1.110 : privatel : ipsec
...
<continue in next slide>
```

OMP Routes state after applying Inbound control policy

```
Controller-1# show support omp rib vroute 10:10.10.110.3/32
```

```
RIB-OUT: (0x7f13192aaf80), RI-ID: 357, Peer: 1.1.1.112 Path-id: 57, Label: 1003, Flags: (0x1) ADV Common (0x7f1313406680) Common Ref: 7
Attribute: (0x7f13134a9200), ROUTE-IPV4, Length: 1184, Ref: 6
Flags: (0x8000c25) WEIGHT TLOC SITE-ID OVERLAY-ID ORIGIN ORIGINATOR
Region-id: 65534, Secondary-Region-id: 65535, Orig-Access-Region-id: 65534, Sub-Region-ID: 0, Pref: 0, Weight: 1, Tag: 0, Stale: 0 Version: 0, Restrict: 0,
on-Demand: 0, Domain: 0, BR-Preference: 0, Affinity-Group-Number:0, MRF-Route-Originator:None , Derived Affinity-group-number: 0 Distance: 0, Site-ID: 110,
Carrier: 0, Query: 0, Gen-ID: 0x0, Border: 0 Overlay: 1 Site-Type: 0 0 0 0
Originator: 1.1.1.110
Origin: Protocol: ospf[4], Sub-Type: intra-area[1], Metric: 11
TLOC: ((nil)) 1.1.1.110 : mpls : ipsec

RIB-OUT: (0x7f13192a9e40), RI-ID: 373, Peer: 1.1.1.112 Path-id: 58, Label: 1003, Flags: (0x1) ADV Common (0x7f1316ac5040) Common Ref: 7
Attribute: (0x7f1317148e00), ROUTE-IPV4, Length: 1184, Ref: 6
Flags: (0x8000c25) WEIGHT TLOC SITE-ID OVERLAY-ID ORIGIN ORIGINATOR
Region-id: 65534, Secondary-Region-id: 65535, Orig-Access-Region-id: 65534, Sub-Region-ID: 0, Pref: 0, Weight: 1, Tag: 0, Stale: 0 Version: 0, Restrict: 0,
on-Demand: 0, Domain: 0, BR-Preference: 0, Affinity-Group-Number:0, MRF-Route-Originator:None , Derived Affinity-group-number: 0 Distance: 0, Site-ID: 110,
Carrier: 0, Query: 0, Gen-ID: 0x0, Border: 0 Overlay: 1 Site-Type: 0 0 0 0
Originator: 1.1.1.110
Origin: Protocol: ospf[4], Sub-Type: intra-area[1], Metric: 11
TLOC: ((nil)) 1.1.1.110 : private1 : ipsec

RIB-OUT: (0x7f13192aad70), RI-ID: 357, Peer: 3.3.3.3 Path-id: 49, Label: 1003, Flags: (0x1) ADV Common (0x7f1313406680) Common Ref: 7
Attribute: (0x7f13134a9200), ROUTE-IPV4, Length: 1184, Ref: 6
Flags: (0x8000c25) WEIGHT TLOC SITE-ID OVERLAY-ID ORIGIN ORIGINATOR
Region-id: 65534, Secondary-Region-id: 65535, Orig-Access-Region-id: 65534, Sub-Region-ID: 0, Pref: 0, Weight: 1, Tag: 0, Stale: 0 Version: 0, Restrict: 0,
on-Demand: 0, Domain: 0, BR-Preference: 0, Affinity-Group-Number:0, MRF-Route-Originator:None , Derived Affinity-group-number: 0 Distance: 0, Site-ID: 110,
Carrier: 0, Query: 0, Gen-ID: 0x0, Border: 0 Overlay: 1 Site-Type: 0 0 0 0
Originator: 1.1.1.110
Origin: Protocol: ospf[4], Sub-Type: intra-area[1], Metric: 11
TLOC: ((nil)) 1.1.1.110 : mpls : ipsec

RIB-OUT: (0x7f13192ac9f0), RI-ID: 373, Peer: 3.3.3.3 Path-id: 56, Label: 1003, Flags: (0x1) ADV Common (0x7f1316ac5040) Common Ref: 7
Attribute: (0x7f1317148e00), ROUTE-IPV4, Length: 1184, Ref: 6
Flags: (0x8000c25) WEIGHT TLOC SITE-ID OVERLAY-ID ORIGIN ORIGINATOR
Region-id: 65534, Secondary-Region-id: 65535, Orig-Access-Region-id: 65534, Sub-Region-ID: 0, Pref: 0, Weight: 1, Tag: 0, Stale: 0 Version: 0, Restrict: 0,
on-Demand: 0, Domain: 0, BR-Preference: 0, Affinity-Group-Number:0, MRF-Route-Originator:None , Derived Affinity-group-number: 0 Distance: 0, Site-ID: 110,
Carrier: 0, Query: 0, Gen-ID: 0x0, Border: 0 Overlay: 1 Site-Type: 0 0 0 0
Originator: 1.1.1.110
Origin: Protocol: ospf[4], Sub-Type: intra-area[1], Metric: 11
TLOC: ((nil)) 1.1.1.110 : private1 : ipsec
```

OMP Routes state after applying Outbound control policy

```
Controller-1# show omp routes vpn 10 advertised | begin VPN
```

VPN	PREFIX	TO PEER
10	10.10.110.3/32	1.1.1.111 1.1.1.112 1.1.1.113 1.1.1.114 1.1.1.115 1.1.1.116 3.3.3.3
10	10.10.110.4/32	1.1.1.111 1.1.1.112 1.1.1.113 1.1.1.114 1.1.1.115 1.1.1.116 3.3.3.3
10	10.10.111.1/32	1.1.1.110 1.1.1.113 1.1.1.114 1.1.1.115 1.1.1.116 3.3.3.3
10	10.10.111.2/32	1.1.1.110 1.1.1.113 1.1.1.114 1.1.1.115 1.1.1.116 3.3.3.3
10	10.10.111.3/32	1.1.1.110 1.1.1.112 1.1.1.113 1.1.1.114 1.1.1.115 1.1.1.116 3.3.3.3
10	10.10.111.4/32	1.1.1.110 1.1.1.112 1.1.1.113 1.1.1.114 1.1.1.115 1.1.1.116 3.3.3.3

OMP Routes state after applying Outbound control policy

```
Controller-1# show support omp rib vroute 10.10.10.111.1/32 rib-out-peer-ip 1.1.1.112
```

```
Looking up vroute 10.10.111.1/32 in 10
```

```
RIB-Entry: (0x7f131609f700) ROUTE-IPV4 Flags: (0x0) , rcv-attr-count 4, adv-attr-count 12
```

```
ri-peer-tree: 0x7f131609f758(4), ro-peer-tree: 0x7f131609f780(12), ro-ri-id-tree: 0x7f131609f7b8(12), Scheduled: 12, Version: 1114, ro-cache-ri-id-tree: 0x7f13160a2e40(2), ro-cache-policy-name-tree: 0x7f13160a2e70(2) VPN-ID: 10, Prefix: 10.10.111.1/32
```

```
Region-Info Region-ID: 65534
```

```
RIB-IN: (0x7f1315bc45e0, prev: (nil), next: 0x7f1315bc4500), Peer: 1.1.1.111, ID: 121, updated: Thu Nov 16 10:57:41 2023
```

```
Path-id: 66, Label: 1003 Affinity Number: 0 TLOC-pref: 0 TLOC-stale: 0 version: 8 (stale: 0) Management-Region: False
```

```
Lost-to-peer: ::, Lost-to-path-id: 0, Loss-Reason: None(0)
```

```
Rcv-Attr: 0x7f13160bcc00, Flags: (0x21) CHOSEN RESOLVED
```

```
Attribute: (0x7f13160bcc00), ROUTE-IPV4, Length: 1184, Ref: 4
```

```
Flags: (0x8000c25) WEIGHT TLOC SITE-ID OVERLAY-ID ORIGIN ORIGINATOR
```

```
Region-id: 65534, Secondary-Region-id: 65535, Orig-Access-Region-id: 65534, Sub-Region-ID: 0, Pref: 0, Weight: 1, Tag: 0, Stale: 0 Version: 0, Restrict: 0, on-Demand: 0, Domain: 0, BR-Preference: 0, Affinity-Group-Number:0, MRF-Route-Originator:None , Derived Affinity-group-number: 0 Distance: 0, Site-ID: 111, Carrier: 0, Query: 0, Gen-ID: 0x0, Border: 0 Overlay: 1 Site-Type: 0 0 0 0
```

```
Originator: 1.1.1.111
```

```
Origin: Protocol: ospf[4], Sub-Type: intra-area[1], Metric: 1
```

```
TLOC: (0x7f1316366a00) 1.1.1.111 : mpls : ipsec
```

```
RIB-IN: (0x7f1315bc4500, prev: 0x7f1315bc45e0, next: 0x7f13153005a0), Peer: 1.1.1.111, ID: 120, updated: Thu Nov 16 10:57:41 2023
```

```
Path-id: 81, Label: 1003 Affinity Number: 0 TLOC-pref: 0 TLOC-stale: 0 version: 8 (stale: 0) Management-Region: False
```

```
Lost-to-peer: 1.1.1.111, Lost-to-path-id: 66, Loss-Reason: TLOC ID(10)
```

```
Rcv-Attr: 0x7f13160bc700, Flags: (0x21) CHOSEN RESOLVED
```

```
Attribute: (0x7f13160bc700), ROUTE-IPV4, Length: 1184, Ref: 4
```

```
Flags: (0x8000c25) WEIGHT TLOC SITE-ID OVERLAY-ID ORIGIN ORIGINATOR
```

```
Region-id: 65534, Secondary-Region-id: 65535, Orig-Access-Region-id: 65534, Sub-Region-ID: 0, Pref: 0, Weight: 1, Tag: 0, Stale: 0 Version: 0, Restrict: 0, on-Demand: 0, Domain: 0, BR-Preference: 0, Affinity-Group-Number:0, MRF-Route-Originator:None , Derived Affinity-group-number: 0 Distance: 0, Site-ID: 111, Carrier: 0, Query: 0, Gen-ID: 0x0, Border: 0 Overlay: 1 Site-Type: 0 0 0 0
```

```
Originator: 1.1.1.111
```

```
Origin: Protocol: ospf[4], Sub-Type: intra-area[1], Metric: 1
```

```
TLOC: (0x7f1316366aa0) 1.1.1.111 : private1 : ipsec
```

```
<continue in next slide>
```

OMP Routes state after applying Outbound control policy

```
Controller-1# show support omp rib vroute 10:10.10.111.1/32 rib-out-peer-ip 1.1.1.112
...
RIB-IN: (0x7f13153005a0, prev: 0x7f1315bc4500, next: 0x7f1315300680), Peer: 3.3.3.3, ID: 149, updated: Sun Nov 12 10:47:01 2023
Path-id: 15, Label: 1003 Affinity Number: 0 TLOC-pref: 0 TLOC-stale: 0 version: 8 (stale: 0) Management-Region: False
Lost-to-peer: 1.1.1.111, Lost-to-path-id: 81, Loss-Reason: Personality(2)
Rcv-Attr: 0x7f13160bcc00, Flags: (0x21) CHOSEN RESOLVED
Attribute: (0x7f13160bcc00), ROUTE-IPV4, Length: 1184, Ref: 4
Flags: (0x8000c25) WEIGHT TLOC SITE-ID OVERLAY-ID ORIGIN ORIGINATOR
Region-id: 65534, Secondary-Region-id: 65535, Orig-Access-Region-id: 65534, Sub-Region-ID: 0, Pref: 0, Weight: 1, Tag: 0, Stale: 0
Version: 0, Restrict: 0, on-Demand: 0, Domain: 0, BR-Preference: 0, Affinity-Group-Number:0, MRF-Route-Originator:None , Derived Affinity-
group-number: 0 Distance: 0, Site-ID: 111, Carrier: 0, Query: 0, Gen-ID: 0x0, Border: 0 Overlay: 1 Site-Type: 0 0 0 0
Originator: 1.1.1.111
Origin: Protocol: ospf[4], Sub-Type: intra-area[1], Metric: 1
TLOC: (0x7f1316366a00) 1.1.1.111 : mpls : ipsec
RIB-IN: (0x7f1315300680, prev: 0x7f13153005a0, next: (nil)), Peer: 3.3.3.3, ID: 150, updated: Sun Nov 12 10:47:01 2023
Path-id: 16, Label: 1003 Affinity Number: 0 TLOC-pref: 0 TLOC-stale: 0 version: 8 (stale: 0) Management-Region: False
Lost-to-peer: 1.1.1.111, Lost-to-path-id: 81, Loss-Reason: Personality(2)
Rcv-Attr: 0x7f13160bc700, Flags: (0x21) CHOSEN RESOLVED
Attribute: (0x7f13160bc700), ROUTE-IPV4, Length: 1184, Ref: 4
Flags: (0x8000c25) WEIGHT TLOC SITE-ID OVERLAY-ID ORIGIN ORIGINATOR
Region-id: 65534, Secondary-Region-id: 65535, Orig-Access-Region-id: 65534, Sub-Region-ID: 0, Pref: 0, Weight: 1, Tag: 0, Stale: 0
Version: 0, Restrict: 0, on-Demand: 0, Domain: 0, BR-Preference: 0, Affinity-Group-Number:0, MRF-Route-Originator:None , Derived Affinity-
group-number: 0 Distance: 0, Site-ID: 111, Carrier: 0, Query: 0, Gen-ID: 0x0, Border: 0 Overlay: 1 Site-Type: 0 0 0 0
Originator: 1.1.1.111
Origin: Protocol: ospf[4], Sub-Type: intra-area[1], Metric: 1
TLOC: (0x7f1316366aa0) 1.1.1.111 : private1 : ipsec
Controller-1#
```

OMP Routes state after applying Outbound control policy

```
Controller-1# show omp routes vpn 10 advertised | begin VPN
```

VPN	PREFIX	TO PEER
10	10.10.110.3/32	1.1.1.111 1.1.1.112 1.1.1.113 1.1.1.114 1.1.1.115 1.1.1.116 3.3.3.3
10	10.10.110.4/32	1.1.1.111 1.1.1.112 1.1.1.113 1.1.1.114 1.1.1.115 1.1.1.116 3.3.3.3
10	10.10.111.1/32	1.1.1.110 1.1.1.113 1.1.1.114 1.1.1.115 1.1.1.116 3.3.3.3
10	10.10.111.2/32	1.1.1.110 1.1.1.113 1.1.1.114 1.1.1.115 1.1.1.116 3.3.3.3
10	10.10.111.3/32	1.1.1.110 1.1.1.112 1.1.1.113 1.1.1.114 1.1.1.115 1.1.1.116 3.3.3.3
10	10.10.111.4/32	1.1.1.110 1.1.1.112 1.1.1.113 1.1.1.114 1.1.1.115 1.1.1.116 3.3.3.3

OMP Routes state after applying Outbound control policy

```
Controller-1# show support omp rib vroute 10.10.10.111.1/32 rib-out-peer-ip 1.1.1.110
```

```
Looking up vroute 10.10.111.1/32 in 10
```

```
RIB-Entry: (0x7f131609f700) ROUTE-IPV4 Flags: (0x0) , recv-attr-count 4, adv-attr-count 12
```

```
ri-peer-tree: 0x7f131609f758(4), ro-peer-tree: 0x7f131609f780(12), ro-ri-id-tree: 0x7f131609f7b8(12), Scheduled: 12, Version: 1114, ro-cache-ri-id-tree: 0x7f13160a2e40(2), ro-cache-policy-name-tree: 0x7f13160a2e70(2) VPN-ID: 10, Prefix: 10.10.111.1/32
```

```
Region-Info Region-ID: 65534
```

```
RIB-IN: (0x7f1315bc45e0, prev: (nil), next: 0x7f1315bc4500), Peer: 1.1.1.111, ID: 121, updated: Thu Nov 16 10:57:41 2023
```

```
Path-id: 66, Label: 1003 Affinity Number: 0 TLOC-pref: 0 TLOC-stale: 0 version: 8 (stale: 0) Management-Region: False
```

```
Lost-to-peer: ::, Lost-to-path-id: 0, Loss-Reason: None(0)
```

```
Rcv-Attr: 0x7f13160bcc00, Flags: (0x21) CHOSEN RESOLVED
```

```
Attribute: (0x7f13160bcc00), ROUTE-IPV4, Length: 1184, Ref: 4
```

```
Flags: (0x8000c25) WEIGHT TLOC SITE-ID OVERLAY-ID ORIGIN ORIGINATOR
```

```
Region-id: 65534, Secondary-Region-id: 65535, Orig-Access-Region-id: 65534, Sub-Region-ID: 0, Pref: 0, Weight: 1, Tag: 0, Stale: 0 Version: 0,
```

```
Restrict: 0, on-Demand: 0, Domain: 0, BR-Preference: 0, Affinity-Group-Number:0, MRF-Route-Originator:None , Derived Affinity-group-number: 0
```

```
Distance: 0, Site-ID: 111, Carrier: 0, Query: 0, Gen-ID: 0x0, Border: 0 Overlay: 1 Site-Type: 0 0 0
```

```
Originator: 1.1.1.111
```

```
Origin: Protocol: ospf[4], Sub-Type: intra-area[1], Metric: 1
```

```
TLOC: (0x7f1316366a00) 1.1.1.111 : mpls : ipsec
```

```
RIB-IN: (0x7f1315bc4500, prev: 0x7f1315bc45e0, next: 0x7f13153005a0), Peer: 1.1.1.111, ID: 120, updated: Thu Nov 16 10:57:41 2023
```

```
Path-id: 81, Label: 1003 Affinity Number: 0 TLOC-pref: 0 TLOC-stale: 0 version: 8 (stale: 0) Management-Region: False
```

```
Lost-to-peer: 1.1.1.111, Lost-to-path-id: 66, Loss-Reason: TLOC ID(10)
```

```
Rcv-Attr: 0x7f13160bc700, Flags: (0x21) CHOSEN RESOLVED
```

```
Attribute: (0x7f13160bc700), ROUTE-IPV4, Length: 1184, Ref: 4
```

```
Flags: (0x8000c25) WEIGHT TLOC SITE-ID OVERLAY-ID ORIGIN ORIGINATOR
```

```
Region-id: 65534, Secondary-Region-id: 65535, Orig-Access-Region-id: 65534, Sub-Region-ID: 0, Pref: 0, Weight: 1, Tag: 0, Stale: 0 Version: 0,
```

```
Restrict: 0, on-Demand: 0, Domain: 0, BR-Preference: 0, Affinity-Group-Number:0, MRF-Route-Originator:None , Derived Affinity-group-number: 0
```

```
Distance: 0, Site-ID: 111, Carrier: 0, Query: 0, Gen-ID: 0x0, Border: 0 Overlay: 1 Site-Type: 0 0 0
```

```
Originator: 1.1.1.111
```

```
Origin: Protocol: ospf[4], Sub-Type: intra-area[1], Metric: 1
```

```
TLOC: (0x7f1316366aa0) 1.1.1.111 : private1 : ipsec
```

```
<snip>
```

```
<continue in next slide>
```

OMP Routes state after applying Outbound control policy

```
Controller-1# show support omp rib vroute 10:10.10.111.1/32 rib-out-peer-ip 1.1.1.110
```

```
..
```

```
RIB-IN: (0x7f13153005a0, prev: 0x7f1315bc4500, next: 0x7f1315300680), Peer: 3.3.3.3, ID: 149, updated: Sun Nov 12 10:47:01 2023
  Path-id: 15, Label: 1003 Affinity Number: 0 TLOC-pref: 0 TLOC-stale: 0 version: 8 (stale: 0) Management-Region: False
  Lost-to-peer: 1.1.1.111, Lost-to-path-id: 81, Loss-Reason: Personality(2)
  Rcv-Attr: 0x7f13160bcc00, Flags: (0x21) CHOSEN RESOLVED
  Attribute: (0x7f13160bcc00), ROUTE-IPV4, Length: 1184, Ref: 4
    Flags: (0x8000c25) WEIGHT TLOC SITE-ID OVERLAY-ID ORIGIN ORIGINATOR
    Region-id: 65534, Secondary-Region-id: 65535, Orig-Access-Region-id: 65534, Sub-Region-ID: 0, Pref: 0, Weight: 1, Tag: 0, Stale: 0 Version:
0, Restrict: 0, on-Demand: 0, Domain: 0, BR-Preference: 0, Affinity-Group-Number:0, MRF-Route-Originator:None , Derived Affinity-group-number: 0
  Distance: 0, Site-ID: 111, Carrier: 0, Query: 0, Gen-ID: 0x0, Border: 0 Overlay: 1 Site-Type: 0 0 0 0
  Originator: 1.1.1.111
  Origin: Protocol: ospf[4], Sub-Type: intra-area[1], Metric: 1
  TLOC: (0x7f1316366a00) 1.1.1.111 : mpls : ipsec
```

```
RIB-IN: (0x7f1315300680, prev: 0x7f13153005a0, next: (nil)), Peer: 3.3.3.3, ID: 150, updated: Sun Nov 12 10:47:01 2023
  Path-id: 16, Label: 1003 Affinity Number: 0 TLOC-pref: 0 TLOC-stale: 0 version: 8 (stale: 0) Management-Region: False
  Lost-to-peer: 1.1.1.111, Lost-to-path-id: 81, Loss-Reason: Personality(2)
  Rcv-Attr: 0x7f13160bc700, Flags: (0x21) CHOSEN RESOLVED
  Attribute: (0x7f13160bc700), ROUTE-IPV4, Length: 1184, Ref: 4
    Flags: (0x8000c25) WEIGHT TLOC SITE-ID OVERLAY-ID ORIGIN ORIGINATOR
    Region-id: 65534, Secondary-Region-id: 65535, Orig-Access-Region-id: 65534, Sub-Region-ID: 0, Pref: 0, Weight: 1, Tag: 0, Stale: 0 Version:
0, Restrict: 0, on-Demand: 0, Domain: 0, BR-Preference: 0, Affinity-Group-Number:0, MRF-Route-Originator:None , Derived Affinity-group-number: 0
  Distance: 0, Site-ID: 111, Carrier: 0, Query: 0, Gen-ID: 0x0, Border: 0 Overlay: 1 Site-Type: 0 0 0 0
  Originator: 1.1.1.111
  Origin: Protocol: ospf[4], Sub-Type: intra-area[1], Metric: 1
  TLOC: (0x7f1316366aa0) 1.1.1.111 : private1 : ipsec
```

```
<snip>
```

```
<continue in next slide>
```

OMP Routes state after applying Outbound control policy

```
Controller-1# show support omp rib vroute 10:10.10.111.1/32 rib-out-peer-ip 1.1.1.110
```

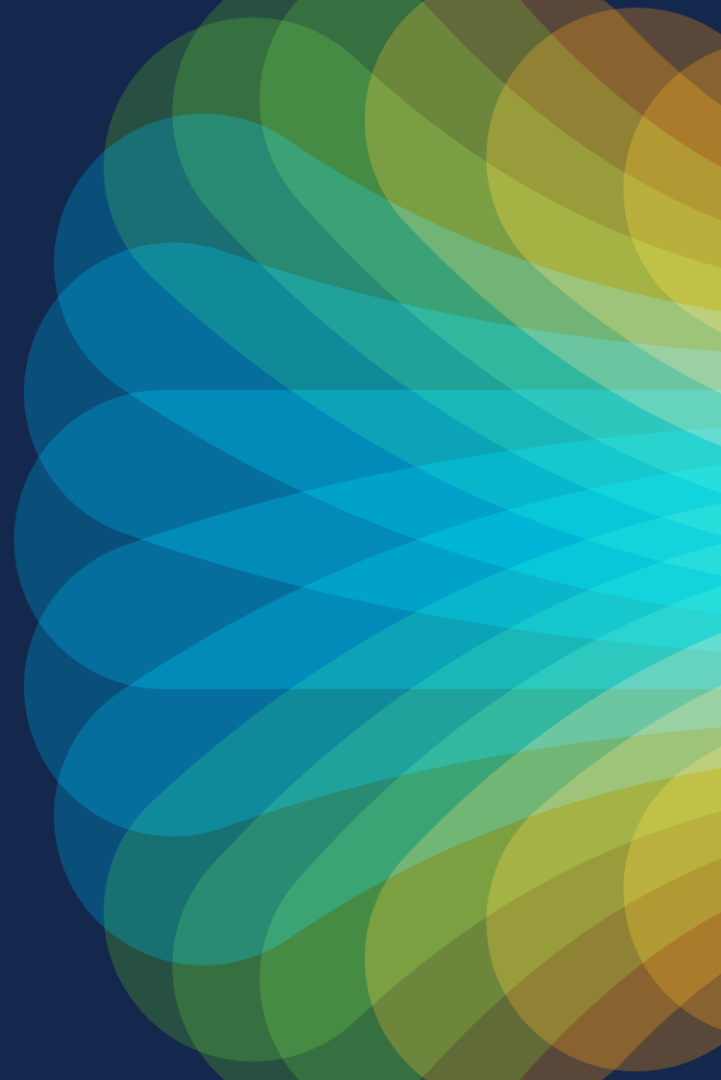
```
..
```

```
RIB-OUT: (0x7f13192aa4a0), RI-ID: 120, Peer: 1.1.1.110 Path-id: 18, Label: 1003, Flags: (0x1) ADV Common (0x7f131342f2c0) Common Ref: 6  
Attribute: (0x7f1313461f00), ROUTE-IPV4, Length: 1184, Ref: 2  
Flags: (0x8000c25) WEIGHT TLOC SITE-ID OVERLAY-ID ORIGIN ORIGINATOR  
Region-id: 65534, Secondary-Region-id: 65535, Orig-Access-Region-id: 65534, Sub-Region-ID: 0, Pref: 0, Weight: 1, Tag: 0, Stale: 0 Version:  
0, Restrict: 0, on-Demand: 0, Domain: 0, BR-Preference: 0, Affinity-Group-Number:0, MRF-Route-Originator:None , Derived Affinity-group-number: 0  
Distance: 0, Site-ID: 111, Carrier: 0, Query: 0, Gen-ID: 0x0, Border: 0 Overlay: 1 Site-Type: 0 0 0 0  
Originator: 1.1.1.111  
Origin: Protocol: ospf[4], Sub-Type: intra-area[1], Metric: 1  
TLOC: ((nil)) 1.1.1.111 : privatel : ipsec
```

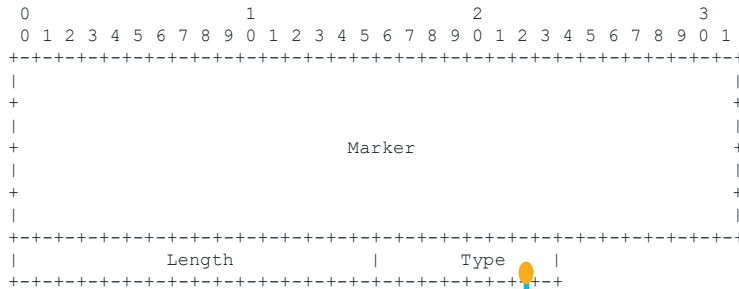
```
RIB-OUT: (0x7f13192aa4d0), RI-ID: 121, Peer: 1.1.1.110 Path-id: 17, Label: 1003, Flags: (0x1) ADV Common (0x7f131342f280) Common Ref: 6  
Attribute: (0x7f1313461a00), ROUTE-IPV4, Length: 1184, Ref: 2  
Flags: (0x8000c25) WEIGHT TLOC SITE-ID OVERLAY-ID ORIGIN ORIGINATOR  
Region-id: 65534, Secondary-Region-id: 65535, Orig-Access-Region-id: 65534, Sub-Region-ID: 0, Pref: 0, Weight: 1, Tag: 0, Stale: 0 Version:  
0, Restrict: 0, on-Demand: 0, Domain: 0, BR-Preference: 0, Affinity-Group-Number:0, MRF-Route-Originator:None , Derived Affinity-group-number: 0  
Distance: 0, Site-ID: 111, Carrier: 0, Query: 0, Gen-ID: 0x0, Border: 0 Overlay: 1 Site-Type: 0 0 0 0  
Originator: 1.1.1.111  
Origin: Protocol: ospf[4], Sub-Type: intra-area[1], Metric: 1  
TLOC: ((nil)) 1.1.1.111 : mpls : ipsec
```

```
Controller-1#
```

OMP and BGP Packet Insights

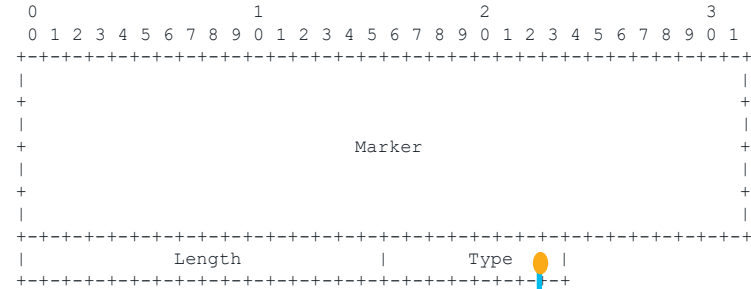
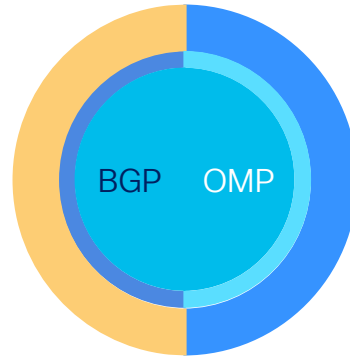


Commonalities between OMP and BGP Packets



BGP Message Header Format

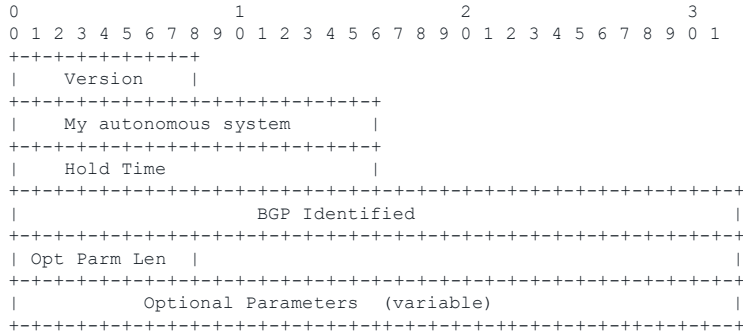
- OPEN
- UPDATE
- NOTIFICATION
- KEEPALIVE



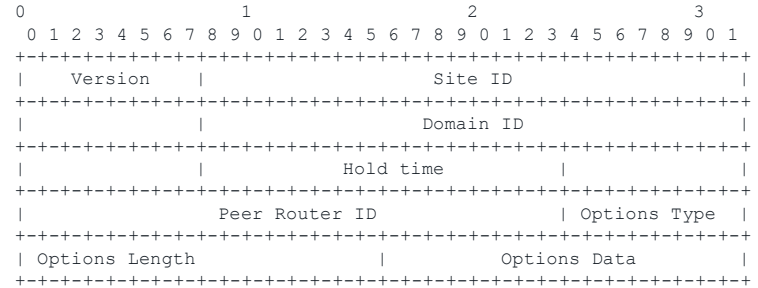
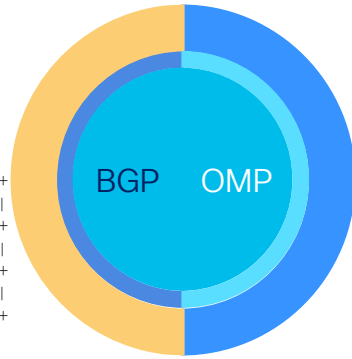
OMP Message Header Format

- HANDSHAKE
- UPDATE
- ALERT
- HELLO
- QUERY
- POLICY
- INFORM

Commonalities between OMP and BGP Packets

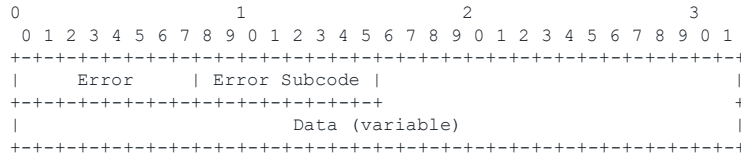


BGP Open Packet Format

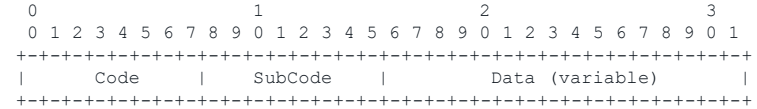
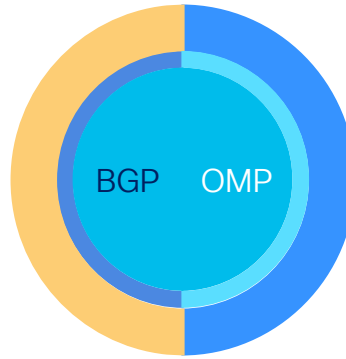


OMP Handshake Packet Format

Commonalities between OMP and BGP Packets



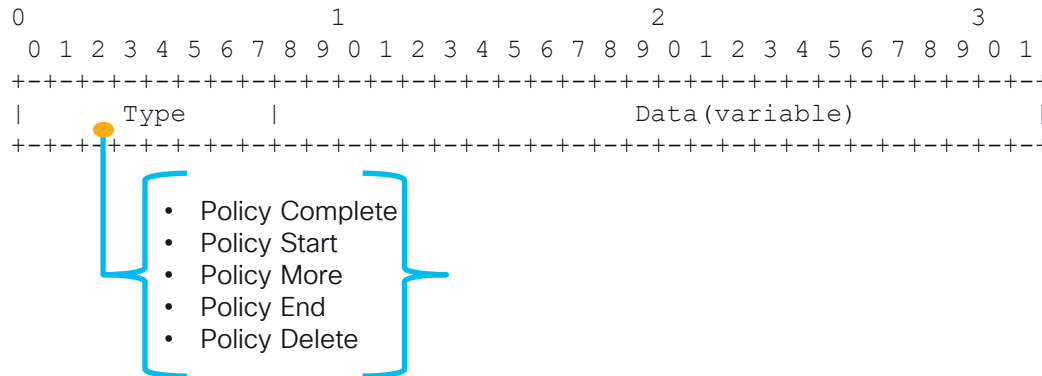
BGP Notification Packet Format



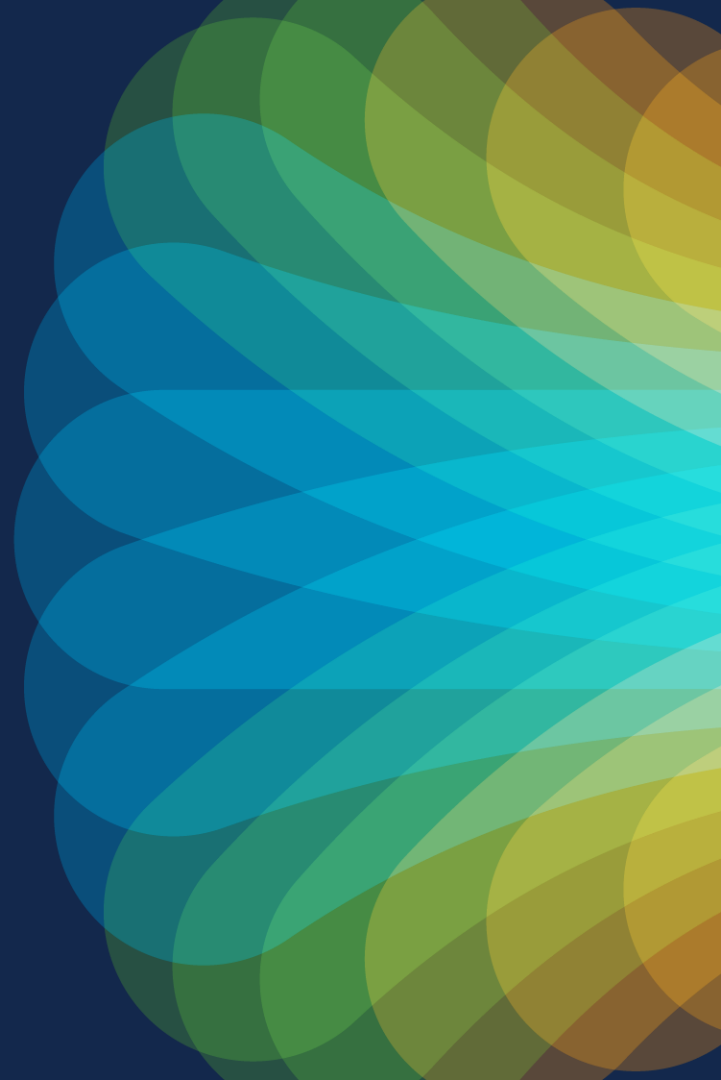
OMP Alert Packet Format

OMP Policy Packet Format

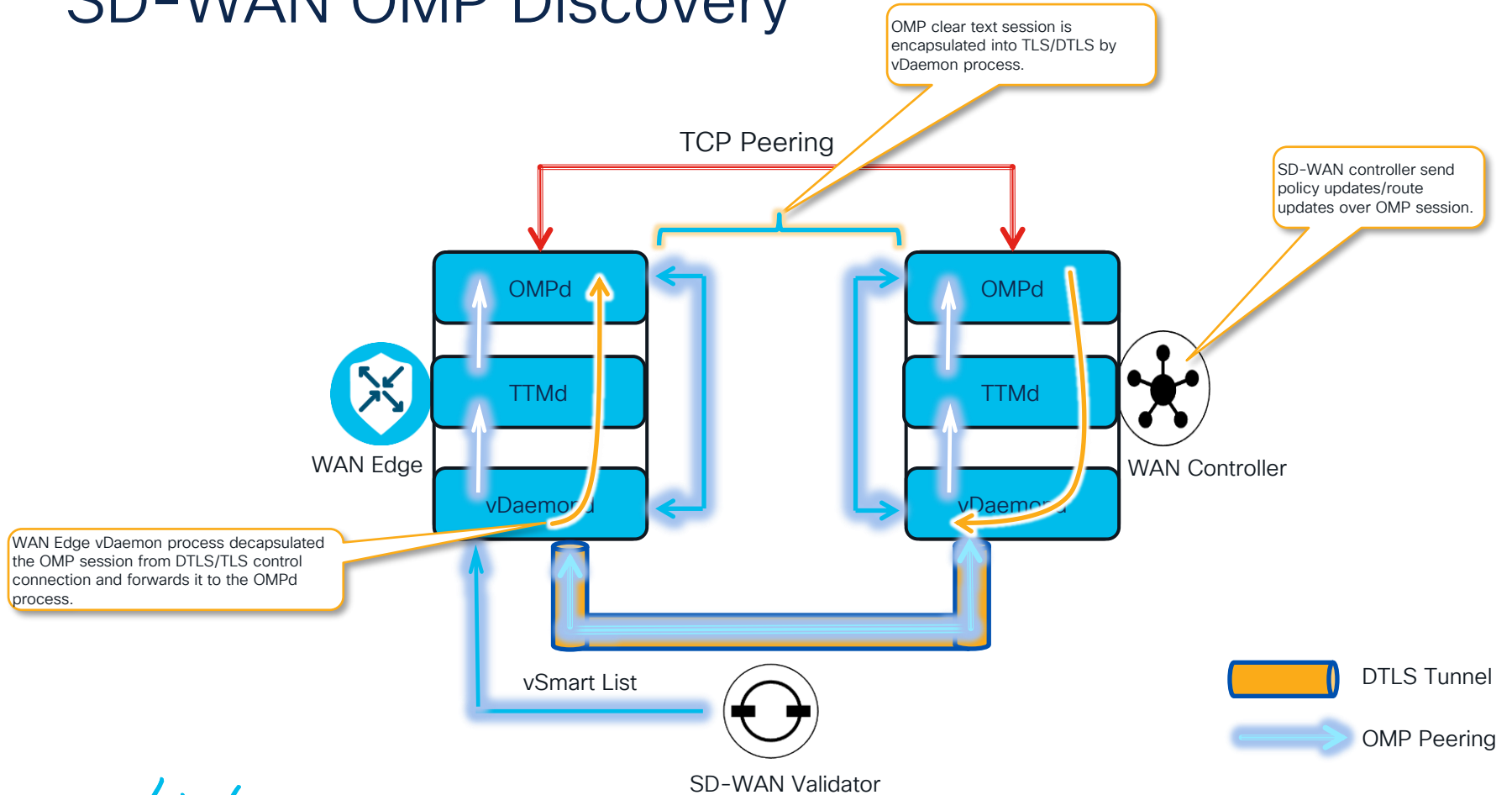
- SD-WAN policies are send in OMP Policy Packet.
- Maximum Policy packet size is **4000 bytes** and if policy is huge then policy will be send in fragments.
- If policy data is **less than 3980** (4000-20 TCP header) bytes, then Type field is set with **Policy Complete**.
- If policy data is **more than 3980 bytes** then it will be send in fragments with **Type field** is set to **Policy Start**, when first packet is send and with **last fragment Type** field is set to **Policy End**.



Walkthrough of OMP Session establishment at the WAN Edge Process level

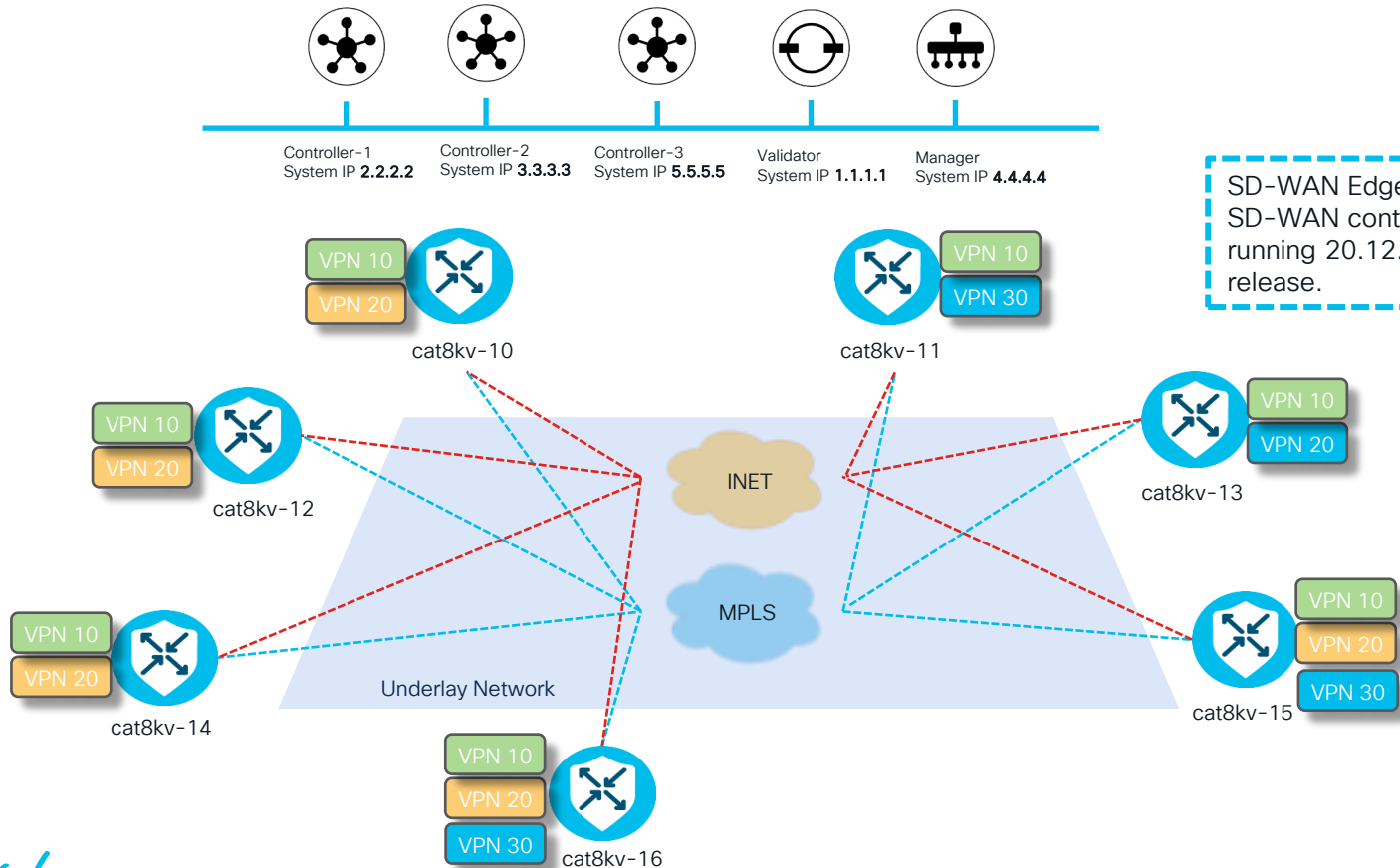


SD-WAN OMP Discovery



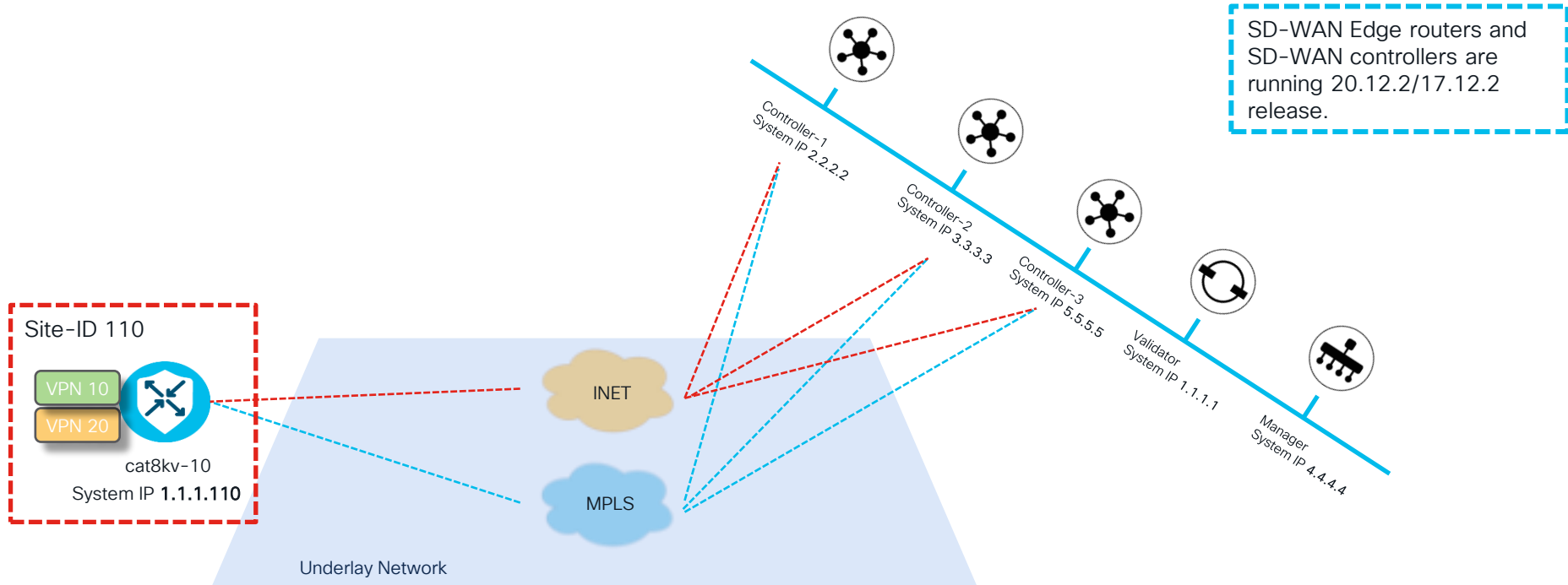
SDWAN OMP Peering Topology

A topic Catalyst SD-WAN demo topology



SD-WAN Edge routers and SD-WAN controllers are running 20.12.2/17.12.2 release.

A topic Catalyst SD-WAN demo topology



Navigating Through Logs During OMP Session Establishment

How can I begin navigating through the OMP logs? 1/3

- From SD-WAN Edge perspective:
 - We can enable OMP debugging by using flags to filter peer OMP negotiation or OMP packets.

```
[no]debug platform software sdwan omp packets "direction both peer-  
address <PEER-IP-ADDRESS> packet-type all"
```

Any SD-WAN debugs do NOT show up in "show debugging" output.

alert
all
cap-update
handshake
hello
inform
policy
query
update

both
received
sent

How can I begin navigating through the OMP logs? 2/3

- From SD-WAN Edge perspective:
 - Starting from the **17.12 release**, we need to enable traces for OMP process along with the module name and then we can view the messages exchanged between SD-WAN controller and SD-WAN Edge devices.
 - `set platform software trace ompd RP active ompd-pkt verbose`
 - `set platform software trace ompd RP active ompd-oper verbose`
 - `set platform software trace ompd RP active ompd-event verbose`
 - By default, all modules for OMP processes are set to the “**Notice**” level.
 - We can verify with the following command if OMP traces are enabled or not.
 - `show platform software trace level ompd rp active`
 - The following command will exhibit what flags are enabled for OMP debugging.
 - `show platform software sdwan RP active internal "omp debug"`
 - To view the OMP debug messages:
 - `show logging process ompd internal`

Note: Once collected all the required logs/traces, set it back to “**Notice**”

How can I begin navigating through the OMP logs? 3/3

- From SD-WAN controller perspective:
 - The following debug can be enabled to view the OMP packet exchange between SD-WAN controllers and WAN-Edges.

```
Controller-1# debug omp
Possible completions:
  best-path      Debug OMP best-path calculation
  cxp            Debug OMP cloudexpress
  events         Debug OMP events
  graceful-restart Debug OMP Graceful Restart
  identity       Debug OMP Identity
  ipcs          Debug OMP IPCs
  packets        Debug OMP packets
  policy         Debug OMP policy
```

- All the debug logs are written by the SD-WAN controller in the following files.

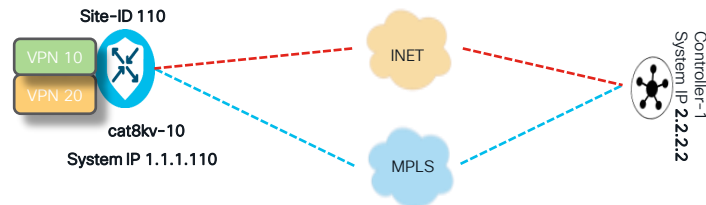
```
Controller-1# vshell
Controller-1:~$ tail -f /var/log/vdebug
```

```
Controller-1# vshell
Controller-1:~$ tail -f /var/log/tmplog/vdebug
```

Let's enable debug and traces !!!

```
cat8kv-10#debug platform software sdwan omp packets "direction both peer-address 2.2.2.2 packet-type all"  
cat8kv-10#set platform software trace ompd RP active ompd-pkt verbose  
cat8kv-10#set platform software trace ompd RP active ompd-oper verbose  
cat8kv-10#set platform software trace ompd RP active ompd-event verbose
```

```
cat8kv-10#show platform software trace level ompd rp active  
Module Name                Trace Level  
-----  
bcrpgc                      Informational  
binos                       Notice  
bipc                        Notice  
btrace                      Notice  
...  
ompd                        Notice  
ompd-bestpath              Notice  
ompd-config                Notice  
ompd-cxp                   Notice  
ompd-event                 Verbose  
ompd-idm                   Notice  
ompd-ipc                   Notice  
ompd-oper                 Verbose  
ompd-pkt                 Verbose  
ompd-policy                Notice  
...  
vipcommon-sql              Notice  
vista                      Notice  
vs_flock                   Notice
```



```
cat8kv-10#show platform software sdwan RP active internal "omp debug"  
Generating output, this might take time, please wait ...  
=====  
DEBUG  
=====  
Events: FALSE level: 0  
IPC: FALSE level: 0, peer_ip: 2.2.2.2  
Packets: TRUE Dir: Both  
Packet Type: HANDSHAKE UPDATE ALERT HELLO QUERY VSMART_CONFIG  
INFORM REGION-ID CAPABILITY-UPDATE
```

SD-WAN Edge debug out for OMP peering establishment 1/3

```
cat8kv-10#set logging marker OMP-PEER-DEBUGGING
cat8kv-10#show logging process ompd internal start marker OMP-PEER-DEBUGGING
Logging display requested on 2023/12/04 22:04:22 (UTC) for Hostname: [cat8kv-10], Model: [C8000V], Version: [17.12.02], SN:
[9LH68E002FF], MD_SN: [SSI130300YK]

Start marker [OMP-PEER-DEBUGGING] at timestamp ["2023/12/04 22:03:27.072967" UTC] found
executing cmd on chassis local ...
Start Marker: OMP-PEER-DEBUGGING
Unified Decoder Library Init .. DONE
Found 1 UTF Streams

2023/12/04 22:03:27.072979383 {smand_R0-0}{255}: Marker Msg: OMP-PEER-DEBUGGING...
2023/12/04 22:03:35.813672896 {ompd_R0-0}{255}: [ompd-event] [24277]: (verbose): Peer 2.2.2.2 is Passive
2023/12/04 22:03:37.204344872 {ompd_R0-0}{255}: [vconfd] [24277]: (note): Enqueued blocking task [Set policy]
2023/12/04 22:03:37.204475342 {ompd_R0-0}{255}: [vconfd] [24277]: (note): Enqueued blocking task [Set Tag-Instances]
2023/12/04 22:03:37.204483438 {ompd_R0-0}{255}: [ompd-event] [24277]: (verbose): Connect retry timer expired for 2.2.2.2
2023/12/04 22:03:37.204484955 {ompd_R0-0}{255}: [ompd-event] [24277]: (verbose): Trying to active connect to 2.2.2.2
2023/12/04 22:03:37.207224532 {ompd_R0-0}{255}: [vconfd] [24277:24354]: (note): Blocking task [Set policy] now scheduled
2023/12/04 22:03:37.207228171 {ompd_R0-0}{255}: [ompd-event] [24277:24354]: (debug): Applying forwarding policy of size: 0 bytes
2023/12/04 22:03:37.209603345 {ompd_R0-0}{255}: [ompd-event] [24277]: (debug): Tenant 0 client port number 62853
2023/12/04 22:03:37.212110905 {ompd_R0-0}{255}: [ompd-event] [24277]: (verbose): Received active connect event 0x80
2023/12/04 22:03:37.212113967 {ompd_R0-0}{255}: [ompd-event] [24277]: (verbose): Received connect from 2.2.2.2
2023/12/04 22:03:37.212143467 {ompd_R0-0}{255}: [ompd-event] [24277]: (verbose): Peer: 2.2.2.2, fd: 60, setsockopt successful!
2023/12/04 22:03:37.212146104 {ompd_R0-0}{255}: [ompd-event] [24277]: (debug): Peer: 2.2.2.2, New Event: Connected curr state: Init
2023/12/04 22:03:37.212152740 {ompd_R0-0}{255}: [ompd-event] [24277]: (debug): Peer: 2.2.2.2 SEND OMP vSmart cfg cap
...
<continue>
```

SD-WAN Edge debug out for OMP peering establishment 2/3

```
2023/12/04 22:03:37.212154949 {ompd_R0-0}{255}: [ompd-pkt] [24277]: (verbose): Sent HANDSHAKE message 152 bytes: peer: 2.2.2.2
2023/12/04 22:03:37.212155955 {ompd_R0-0}{255}: [ompd-pkt] [24277]: (verbose): Version: 1 Holdtime: 60 site: 110 domain: 1
2023/12/04 22:03:37.212159867 {ompd_R0-0}{255}: [ompd-pkt] [24277]: (verbose): Peer-ID: 110.1.1.1 Options-Length: 117
2023/12/04 22:03:37.212161079 {ompd_R0-0}{255}: [ompd-pkt] [24277]: (verbose): Option-Type: Capabilities(2) Option-Length: 115
2023/12/04 22:03:37.212162361 {ompd_R0-0}{255}: [ompd-pkt] [24277]: (verbose): Capability: Multi-Protocol Length: 4
2023/12/04 22:03:37.212162983 {ompd_R0-0}{255}: [ompd-pkt] [24277]: (verbose): Value:
2023/12/04 22:03:37.212164292 {ompd_R0-0}{255}: [ompd-pkt] [24277]: (verbose): AFI: ipv4(1) SAFI: tloc(2) VERSION:0
2023/12/04 22:03:37.212165207 {ompd_R0-0}{255}: [ompd-pkt] [24277]: (verbose): Capability: Multi-Protocol Length: 4
2023/12/04 22:03:37.212165748 {ompd_R0-0}{255}: [ompd-pkt] [24277]: (verbose): Value:
2023/12/04 22:03:37.212166571 {ompd_R0-0}{255}: [ompd-pkt] [24277]: (verbose): AFI: ipv4(1) SAFI: service(3) VERSION:0
2023/12/04 22:03:37.212167399 {ompd_R0-0}{255}: [ompd-pkt] [24277]: (verbose): Capability: Multi-Protocol Length: 4
2023/12/04 22:03:37.212168177 {ompd_R0-0}{255}: [ompd-pkt] [24277]: (verbose): Value:
2023/12/04 22:03:37.212169113 {ompd_R0-0}{255}: [ompd-pkt] [24277]: (verbose): AFI: ipv6(2) SAFI: service(3) VERSION:0
2023/12/04 22:03:37.212197442 {ompd_R0-0}{255}: [ompd-pkt] [24277]: (verbose): Capability: Multi-Protocol Length: 4
2023/12/04 22:03:37.212197981 {ompd_R0-0}{255}: [ompd-pkt] [24277]: (verbose): Value:
2023/12/04 22:03:37.212198479 {ompd_R0-0}{255}: [ompd-pkt] [24277]: (verbose): AFI: ipv4(1) SAFI: vroute(1) VERSION:0
2023/12/04 22:03:37.212198753 {ompd_R0-0}{255}: [ompd-pkt] [24277]: (verbose): Capability: Multi-Protocol Length: 4
2023/12/04 22:03:37.212199009 {ompd_R0-0}{255}: [ompd-pkt] [24277]: (verbose): Value:
2023/12/04 22:03:37.212199306 {ompd_R0-0}{255}: [ompd-pkt] [24277]: (verbose): AFI: ipv6(2) SAFI: vroute(1) VERSION:0
2023/12/04 22:03:37.212199588 {ompd_R0-0}{255}: [ompd-pkt] [24277]: (verbose): Capability: Multi-Protocol Length: 4
2023/12/04 22:03:37.212199842 {ompd_R0-0}{255}: [ompd-pkt] [24277]: (verbose): Value:
...
2023/12/04 22:03:37.212208723 {ompd_R0-0}{255}: [ompd-pkt] [24277]: (verbose): Capability: Identity Length: 4 Value: vEdge
2023/12/04 22:03:37.212216872 {ompd_R0-0}{255}: [ompd-pkt] [24277]: (verbose): Capability: Graceful-Restart Length: 4 Value: 43200
...
2023/12/04 21:34:50.939057819 {ompd_R0-0}{255}: [ompd-pkt] [24277]: (verbose):
2023/12/04 21:34:50.939177708 {ompd_R0-0}{255}: [ompd-event] [24277]: (verbose): Peer: 2.2.2.2, New State: Handshake
...
<continue>
```

SD-WAN Edge debug out for OMP peering establishment 3/3

```
2023/12/04 22:03:37.219527088 {ompd_R0-0}{255}: [ompd-pkt] [24277]: (verbose): Received HANDSHAKE message 137 bytes: peer: 2.2.2.2
2023/12/04 22:03:37.219532023 {ompd_R0-0}{255}: [ompd-pkt] [24277]: (verbose): Version: 1 Holdtime: 300 site: 100 domain: 1
2023/12/04 22:03:37.219536623 {ompd_R0-0}{255}: [ompd-pkt] [24277]: (verbose): Peer-ID: 2.2.2.2 Options-Length: 102
2023/12/04 22:03:37.219537834 {ompd_R0-0}{255}: [ompd-pkt] [24277]: (verbose): Option-Type: Capabilities(2) Option-Length: 100
2023/12/04 22:03:37.219538794 {ompd_R0-0}{255}: [ompd-pkt] [24277]: (verbose): Capability: Multi-Protocol Length: 4
2023/12/04 22:03:37.219539724 {ompd_R0-0}{255}: [ompd-pkt] [24277]: (verbose): Value:
2023/12/04 22:03:37.219541061 {ompd_R0-0}{255}: [ompd-pkt] [24277]: (verbose): AFI: ipv4(1) SAFI: tloc(2) VERSION:0
2023/12/04 22:03:37.219542019 {ompd_R0-0}{255}: [ompd-pkt] [24277]: (verbose): Capability: Multi-Protocol Length: 4
2023/12/04 22:03:37.219542563 {ompd_R0-0}{255}: [ompd-pkt] [24277]: (verbose): Value:
2023/12/04 22:03:37.219543201 {ompd_R0-0}{255}: [ompd-pkt] [24277]: (verbose): AFI: ipv4(1) SAFI: service(3) VERSION:0
2023/12/04 22:03:37.219543707 {ompd_R0-0}{255}: [ompd-pkt] [24277]: (verbose): Capability: Multi-Protocol Length: 4
2023/12/04 22:03:37.219544246 {ompd_R0-0}{255}: [ompd-pkt] [24277]: (verbose): Value:
2023/12/04 22:03:37.219544894 {ompd_R0-0}{255}: [ompd-pkt] [24277]: (verbose): AFI: ipv6(2) SAFI: service(3) VERSION:0
2023/12/04 22:03:37.219545407 {ompd_R0-0}{255}: [ompd-pkt] [24277]: (verbose): Capability: Multi-Protocol Length: 4
2023/12/04 22:03:37.219545799 {ompd_R0-0}{255}: [ompd-pkt] [24277]: (verbose): Value:
2023/12/04 22:03:37.219546438 {ompd_R0-0}{255}: [ompd-pkt] [24277]: (verbose): AFI: ipv4(1) SAFI: vroute(1) VERSION:0
2023/12/04 22:03:37.219546717 {ompd_R0-0}{255}: [ompd-pkt] [24277]: (verbose): Capability: Multi-Protocol Length: 4
...
2023/12/04 22:03:37.220203220 {ompd_R0-0}{255}: [ompd-pkt] [24277]: (verbose): Capability: Identity Length: 4 Value: vSmart
2023/12/04 22:03:37.220203883 {ompd_R0-0}{255}: [ompd-pkt] [24277]: (verbose): Capability: Graceful-Restart Length: 4 Value: 43200
...
2023/12/04 22:03:37.220234596 {ompd_R0-0}{255}: [ompd-event] [24277]: (debug): Peer: 2.2.2.2, New Event: Handshake Recv curr state: Handshake
2023/12/04 22:03:37.220240370 {ompd_R0-0}{255}: [ompd-pkt] [24277]: (verbose): Sent HELLO message of length: 27: peer: 2.2.2.2 (in: 156, out: 0)
2023/12/04 22:03:37.220324160 {ompd_R0-0}{255}: [ompd-pkt] [24277]: (verbose): Received HELLO message of length: 19: peer: 2.2.2.2 (in: 156, out: 0)
2023/12/04 22:03:37.220426062 {ompd_R0-0}{255}: [ompd-event] [24277]: (verbose): Peer: 2.2.2.2, New State: Up
```

SD-WAN Edge OMP peering is UP !!!

```
cat8kv-10#show sdwan omp peers
```

```
R -> routes received
```

```
I -> routes installed
```

```
S -> routes sent
```

TENANT ID	PEER	TYPE	DOMAIN ID	OVERLAY ID	SITE ID	REGION ID	STATE	UPTIME	R/I/S
0	2.2.2.2	vsmart	1	1	100	None	up	0:00:35:11	200/200/40
0	3.3.3.3	vsmart	1	1	100	None	up	10:07:46:05	200/0/40

- Use the following command on SD-WAN Edge device to see all the negotiated parameters , advertised TLOC's, address families etc.. with SD-WAN controller.
 - `show platform software sdwan RP active internal "omp peer peer-ip <PEER-IP>"`
- Same command is available also on SD-WAN controller.
 - `show support omp peer peer-ip <PEER-IP>`

SD-WAN Edge OMP peer output

```
cat8kv-10#show platform software sdwan RP active internal "omp peer peer-ip 2.2.2.2"
```

```
Generating output, this might take time, please wait ...
```

```
=====
PEERS for CONTEXT 1.1.1.110, Tenant id: 0
=====
Local address: 1.1.1.110
Looking up Peer: 2.2.2.2
Peer: 2.2.2.2 (0x55da7cdebee8), Type: vSmart, Site: 100, Region-id-set: None, Domain: 1, Overlay: 1, Legit: yes
  State: Up, version: 1, Control-Up: yes, Staging: no, flags: 0x21
  Multithreading- down: no, move-marker: no, update-gen: no, work-queue: no, needs_upd:0x0
  buffer ev: 0x0x55da7ce98dd8, src_port 62853
  fd: 60
  Hello timer: Enabled (e: 36, c: 100, md: 100 lmd: 0) Hold timer: Enabled (e: 237 v: 300 c: 300) Expiry-Alert-Send-Fail-Count: 0
  Connect retry: Disabled (e: -1 v: 2 c: 2) Adv. timer: Enabled (e: 1 v: 1 c: 1)
  Down-pending: Disabled (e: -1 v: 1 c: 1)
  EOR interval: 300 EOR timer: Disabled (e: -1 v: 300)

  Force-Send interval: 2 Force-Send timer: Disabled (e: -1 v: 2)

  Rcv cap: Identity MP GR Refresh Security Overlay
  Neg cap: Identity MP GR Refresh Security Overlay
  Rcv afi-safi: TLOC-IPV4 SRVC-IPV4 SRVC-IPV6 ROUTE-IPV4 ROUTE-IPV6 MCAST-IPV4 (2) LINK CXP (2) IDENTITY-USER IDENTITY-IPV4-TO-USER
IDENTITY-IPV6-TO-USER IDENTITY-IPV4-TO-SGT IDENTITY-IPV6-TO-SGT
  Neg afi-safi: TLOC-IPV4 SRVC-IPV4 SRVC-IPV6 ROUTE-IPV4 ROUTE-IPV6 MCAST-IPV4 (2) LINK CXP (2) IDENTITY-USER IDENTITY-IPV4-TO-USER
IDENTITY-IPV6-TO-USER IDENTITY-IPV4-TO-SGT IDENTITY-IPV6-TO-SGT
  GR-enabled: Enabled, My GR interval: 43200 GR timer: Disabled (e: -1 v: 43200 c: 43200)
  Enter gr: 1, Exit gr: 1, GR mode: FALSE
  site-pol: None route-pol-in: None route-pol-out: None data-pol-in: None
  data-pol-out: None pfr-pol: None mem-pol: None cflowd:None
  UP time: Mon Dec 4 22:03:37 2023
```

```
<snip>
```

SD-WAN Edge OMP peer output

Needs Update gen: 0x0

TLOC-IPV4:

EOR - TX: 1 RX: 1
Browse-Done: 1 Force-Send: 0
received: 12 installed: 12 **sent: 2**
ri-cleanup: 72 ro-cleanup: 12 ri-reeval: 0 reeval: 0
marker-reset: 0 routes-browse: 497 ri-browsed: 497 te-changed: 0
ctx-rib-version: 680 peer-ro-version: 680

ROUTE-IPV4:

EOR - TX: 1 RX: 1
Browse-Done: 1 Force-Send: 0
received: 100 installed: 100 **sent: 20**
ri-cleanup: 600 ro-cleanup: 120 ri-reeval: 0 reeval: 0
marker-reset: 0 routes-browse: 765 ri-browsed: 2470 te-changed: 0
ctx-rib-version: 1190 peer-ro-version: 1190

Updates pending: No, Withdraws pending: No

ROUTE-IPV6:

EOR - TX: 1 RX: 1
Browse-Done: 1 Force-Send: 0
received: 100 installed: 100 **sent: 20**
ri-cleanup: 600 ro-cleanup: 120 ri-reeval: 0 reeval: 0
marker-reset: 0 routes-browse: 765 ri-browsed: 2470 te-changed: 0
ctx-rib-version: 1190 peer-ro-version: 1190

Updates pending: No, Withdraws pending: No

<snip>

SD-WAN Controller OMP peer output

```
Controller-1#show support omp peer peer-ip 1.1.1.110
```

```
=====
PEERS for CONTEXT 2.2.2.2
=====
Local address: 2.2.2.2
Looking up Peer: 1.1.1.110
Peer: 1.1.1.110 (0x7fbcc8dbf400), Type: vEdge, Site: 110, Region-id-set: None, Domain: 1, Overlay: 1, Legit: yes
  State: Up, version: 1, Control-Up: yes, Staging: no, flags: 0x21, Peer ID: 4
  CAP: TLOC color supported bitmap: 0x20004, TLOC color supported list: mpls private1
  CAP: BR: no, TGW: no MGW: no MGMT-REGION: no
  Multithreading- down: no, move-marker: no, update-gen: no, work-queue: no, needs_upd: 0x0
  buffer ev: 0x0x7fbd44a71f00
  fd: 29
  Hello timer: Enabled (e: 38, c: 100, md: 100 lmd: 0) Hold timer: Enabled (e: 239 v: 300 c: 300) Expiry-Alert-Send-Fail-Count: 0
  Connect retry: Disabled (e: -1 v: 2 c: 2) Adv. timer: Enabled (e: 1 v: 1 c: 1)
  Down-pending: Disabled (e: -1 v: 1 c: 1)
  EOR interval: 300 EOR timer: Disabled (e: -1 v: 300)

  Force-Send interval: 2 Force-Send timer: Disabled (e: -1 v: 2)

  Rcv cap: Identity MP GR Refresh Security Overlay Network-Identity VSmart-Configuration
  Neg cap: Identity MP GR Refresh Security Overlay Network-Identity VSmart-Configuration
  Rcv afi-safi: TLOC-IPV4 SRVC-IPV4 SRVC-IPV6 ROUTE-IPV4 ROUTE-IPV6 MCAST-IPV4 (2) LINK CXP (2) IDENTITY-USER IDENTITY-IPV4-TO-USER IDENTITY-IPV6-TO-USER IDENTITY-IPV4-TO-SGT IDENTITY-IPV6-TO-SGT
  Neg afi-safi: TLOC-IPV4 SRVC-IPV4 SRVC-IPV6 ROUTE-IPV4 ROUTE-IPV6 MCAST-IPV4 (2) LINK CXP (2) IDENTITY-USER IDENTITY-IPV4-TO-USER IDENTITY-IPV6-TO-USER IDENTITY-IPV4-TO-SGT IDENTITY-IPV6-TO-SGT
  GR-enabled: Enabled, My GR interval: 43200 GR timer: Disabled (e: -1 v: 43200 c: 43200)
  Enter gr: 0, Exit gr: 0, GR mode: FALSE
  site-pol: None route-pol-in: None route-pol-out: None data-pol-in: None
    data-pol-out: None pfr-pol: None mem-pol: None cflowd:None
  UP time: Mon Dec 4 23:03:45 2023
  Last DOWN time: Mon Dec 4 23:03:44 2023
  Down Event: Alert, Err code: Invalid, Subcode: 0, Down-pend: no
  UP: 4, DOWN: 3, CONN: 0
```

```
<SNIP>
```

SD-WAN Controller OMP peer output

```
Route policy in timer: Disabled (-1)
Membership policy timer: Disabled (-1)
Forwarding policy timer: Disabled (-1), Push status: 0
Forwarding tag-instances timer: Disabled (-1), Push status: 0
  Policy received: Complete
Forwarding policy len: 0
Statistics:

Needs Update gen: 0x0
TLOC-IPV4:
EOR - TX: 1 RX: 1
Browse-Done: 1 Force-Send: 0
received: 2 installed: 2 sent: 12
ri-cleanup: 6 ro-cleanup: 36 ri-reeval: 32 reeval: 208
marker-reset: 1 routes-browse: 96 ri-browsed: 96 te-changed: 0
ctx-rib-version: 240 peer-ro-version: 240
  Updates pending: No, Withdraws pending: No
ROUTE-IPV4:
EOR - TX: 1 RX: 1
Browse-Done: 1 Force-Send: 0
received: 20 installed: 0 sent: 100
ri-cleanup: 60 ro-cleanup: 300 ri-reeval: 80 reeval: 1068
marker-reset: 29 routes-browse: 576 ri-browsed: 2100 te-changed: 0
ctx-rib-version: 599 peer-ro-version: 599
ro-cache-hit: 0 ro-cache-miss: 0
  Updates pending: No, Withdraws pending: No
ROUTE-IPV6:
EOR - TX: 1 RX: 1
Browse-Done: 1 Force-Send: 0
received: 20 installed: 0 sent: 100
ri-cleanup: 60 ro-cleanup: 300 ri-reeval: 80 reeval: 900
marker-reset: 29 routes-browse: 480 ri-browsed: 1740 te-changed: 0
ctx-rib-version: 355 peer-ro-version: 355
ro-cache-hit: 0 ro-cache-miss: 0
  Updates pending: No, Withdraws pending: No
```

<snip>

How can we see the OMP policy packets?

- Following debugs needs to be enable on SD-WAN controller to see the OMP policy packet exchanged, while pushing SD-WAN policies.
 - `debug omp events level high`
 - `debug omp policy level high`
 - `debug omp policy direction both`
 - `debug omp packets packet-type policy direction both`
- The following traces must be enabled on WAN edge to see the exchange of OMP policy packets while SD-WAN policies are being pushed.
 - `set platform software trace ompd RP active ompd-policy verbose`
 - `set platform software trace ompd RP active ompd-pkt verbose`
 - `set platform software trace ompd Rp active ompd-event verbose`
- We can verify with the following command if OMP traces are enabled or not.
 - `show platform software trace level ompd rp active`

How to read OMP policy packet logs on controller 1/3?

```
Controller1# vshell
Controller1:~$ tail -f /var/log/vdebug
Jan  5 22:51:15 Controller1 OMPD[2113]: omp_pkt_encode_policy[701]: Peer 1.1.1.110, total fragments: 4, pol_size: 14085
Jan  5 22:51:15 Controller1 OMPD[2113]: omp_pkt_encode_policy[734]: Peer 1.1.1.110, type: START Current fragments: 1, frag_len: 1
cfg_sub_type = 3979
Jan  5 22:51:15 Controller1 OMPD[2113]: omp_debug_pkt_policy[1621]: Sent POLICY message 4000 bytes: peer: 1.1.1.110
Jan  5 22:51:15 Controller1 OMPD[2113]: omp_debug_pkt_policy[1636]: length: 4000, pol_sub_type: STARTcfg_sub_type = POLICY
Jan  5 22:51:15 Controller1 OMPD[2113]: omp_debug_pkt_policy[1640]: Forwarding (data + pfr): <data-policy>
<name>_ALL-VPN_Data-Policy-QoS-Demo</name>
<vpn-list>
  <name>ALL-VPN</name>
  <sequence>
    <seq-value>1</seq-value>
    <match>
      <source-data-prefix-list>Data-Prefix-Trex-Site-201</source-data-prefix-list>
      <destination-data-prefix-list>Data-Prefix-Trex-Site-200</destination-data-prefix-list>
      <source-port>63000</source-port>
      <destination-port>53000</destination-port>
    </match>
    <action>
      <action-value>accept</action-value>
      <count>DP-Port-63000_-1125422396</count>
      <log/>
      <set>
        <forwarding-class>REAL-TIME</forwarding-class>
      </set>
    </action>
  ...
  <continue>
```

How to read OMP policy packet logs on controller 2/3?

```
Jan  5 22:51:15 Controller1 OMPD[2113]: omp_debug_pkt_policy[1641]:
Jan  5 22:51:15 Controller1 OMPD[2113]: omp_pkt_encode_policy[734]: Peer 1.1.1.110, type: MORE Current fragments: 2, frag_len: 1
cfg_sub_type = 3979
Jan  5 22:51:15 Controller1 OMPD[2113]: omp_debug_pkt_policy[1621]: Sent POLICY message 4000 bytes: peer: 1.1.1.110
Jan  5 22:51:15 Controller1 OMPD[2113]: omp_debug_pkt_policy[1636]: length: 4000, pol_sub_type: MOREcfg_sub_type = POLICY
Jan  5 22:51:15 Controller1 OMPD[2113]: omp_debug_pkt_policy[1640]: Forwarding (data + pfr): lass>
</action>
  </sequence>
</sequence>
  <seq-value>21</seq-value>
  <match>
    <source-data-prefix-list>Data-Prefix-Trex-Site-201</source-data-prefix-list>
    <destination-data-prefix-list>Data-Prefix-Trex-Site-200</destination-data-prefix-list>
    <source-port>6200</source-port>
    <destination-port>5200</destination-port>
    <protocol>17</protocol>
  </match>
...
Jan  5 22:51:15 Controller1 OMPD[2113]: omp_debug_pkt_policy[1641]:
Jan  5 22:51:15 Controller1 OMPD[2113]: omp_pkt_encode_policy[734]: Peer 1.1.1.110, type: MORE Current fragments: 3, frag_len: 1
cfg_sub_type = 3979
Jan  5 22:51:15 Controller1 OMPD[2113]: omp_debug_pkt_policy[1621]: Sent POLICY message 4000 bytes: peer: 1.1.1.110
Jan  5 22:51:15 Controller1 OMPD[2113]: omp_debug_pkt_policy[1636]: length: 4000, pol_sub_type: MOREcfg_sub_type = POLICY
Jan  5 22:51:15 Controller1 OMPD[2113]: omp_debug_pkt_policy[1640]: Forwarding (data + pfr): <color>
  <color-name>private1</color-name>
  <dscp>8</dscp>
</color>
</app-probe-class>
...
<continue>
```

How to read OMP policy packet logs on controller 3/3?

```
Jan  5 22:51:15 Controller1 OMPD[2113]: omp_debug_pkt_policy[1641]:
Jan  5 22:51:15 Controller1 OMPD[2113]: omp_pkt_encode_policy[734]: Peer 1.1.1.110, type: END Current fragments: 4, frag_len: 1
cfg_sub_type = 2148
Jan  5 22:51:15 Controller1 OMPD[2113]: omp_debug_pkt_policy[1621]: Sent POLICY message 2169 bytes: peer: 1.1.1.110
Jan  5 22:51:15 Controller1 OMPD[2113]: omp_debug_pkt_policy[1636]: length: 2169, pol_sub_type: ENDcfg_sub_type = POLICY
Jan  5 22:51:15 Controller1 OMPD[2113]: omp_debug_pkt_policy[1640]: Forwarding (data + pfr): sh</name>
  </app>
  <app>
    <name>live_storage</name>
  </app>
  <app>
...
  <app>
    <name>yammer</name>
  </app>
</app-list>
</lists>ame>google_groups</n
Jan  5 22:51:15 Controller1 OMPD[2113]: omp_debug_pkt_policy[1641]:
```

How to read OMP policy packet logs on WAN-Edge?

```
cat8kv-10#show logging process ompd internal
```

```
...
2024/01/06 09:55:34.296078349 {ompd_R0-0}{255}: [ompd-pkt] [22843]: (verbose): Received POLICY message 4000 bytes: peer: 2.2.2.2
2024/01/06 09:55:34.296081770 {ompd_R0-0}{255}: [ompd-pkt] [22843]: (verbose): length: 4000, type: START
2024/01/06 09:55:34.296085881 {ompd_R0-0}{255}: [ompd-pkt] [22843]: (verbose): Forwarding (data + pfr): <data-policy>
  <name>_ALL-VPN_Data-Policy-QoS-Demo</name>
  <vpn-list>
...
2024/01/06 09:55:34.296156109 {ompd_R0-0}{255}: [ompd-event] [22843]: (verbose): Partial message for peer 2.2.2.2, bailing
2024/01/06 09:55:34.296175046 {ompd_R0-0}{255}: [ompd-pkt] [22843]: (verbose): Received POLICY message 4000 bytes: peer: 2.2.2.2
2024/01/06 09:55:34.296175829 {ompd_R0-0}{255}: [ompd-pkt] [22843]: (verbose): length: 4000, type: MORE
2024/01/06 09:55:34.296178825 {ompd_R0-0}{255}: [ompd-pkt] [22843]: (verbose): Forwarding (data + pfr): lass>
  </action>
  </sequence>
  <sequence>
...
2024/01/06 09:55:34.305498345 {ompd_R0-0}{255}: [ompd-event] [22843]: (verbose): Partial message for peer 2.2.2.2, bailing
2024/01/06 09:55:34.306100223 {ompd_R0-0}{255}: [ompd-pkt] [22843]: (verbose): Received POLICY message 4000 bytes: peer: 2.2.2.2
2024/01/06 09:55:34.306102732 {ompd_R0-0}{255}: [ompd-pkt] [22843]: (verbose): length: 4000, type: MORE
2024/01/06 09:55:34.306107332 {ompd_R0-0}{255}: [ompd-pkt] [22843]: (verbose): Forwarding (data + pfr): <color>
  <color-name>private1</color-name>
  <dscp>8</dscp>
...
2024/01/06 09:55:34.306125175 {ompd_R0-0}{255}: [ompd-pkt] [22843]: (verbose):
2024/01/06 09:55:34.306149702 {ompd_R0-0}{255}: [ompd-pkt] [22843]: (verbose): Received POLICY message 2169 bytes: peer: 2.2.2.2
2024/01/06 09:55:34.306150713 {ompd_R0-0}{255}: [ompd-pkt] [22843]: (verbose): length: 2169, type: END
2024/01/06 09:55:34.306154298 {ompd_R0-0}{255}: [ompd-pkt] [22843]: (verbose): Forwarding (data + pfr): sh</name>
  </app>
  <app>
..
2024/01/06 09:55:34.306163615 {ompd_R0-0}{255}: [ompd-pkt] [22843]: (verbose):
```

How can we verify if policy is being applied correctly?

```
Controller1#show support omp peer peer-ip 1.1.1.112
```

```
=====
PEERS for CONTEXT 2.2.2.2
=====
Local address: 2.2.2.2
Looking up Peer: 1.1.1.112
Peer: 1.1.1.112 (0x7fd8cd031800), Type: vEdge, Site: 112, Region-id-set: None, Domain: 1, Overlay: 1, Legit: yesa
  State: Up, version: 1, Control-Up: yes, Staging: no, flags: 0x21, Peer ID: 4
  CAP: TLOC color supported bitmap: 0x20004, TLOC color supported list: mpls private1
  CAP: BR: no, TGW: no MGW: no MGMT-REGION: no
  Multithreading- down: no, move-marker: no, update-gen: no, work-queue: no, needs_upd: 0x0
  buffer ev: 0x0x7fd949672400
  fd: 29
  Hello timer: Enabled (e: 73, c: 100, md: 100 lmd: 0) Hold timer: Enabled (e: 225 v: 300 c: 300) Expiry-Alert-Send-Fail-Count: 0
  Connect retry: Disabled (e: -1 v: 2 c: 2) Adv. timer: Enabled (e: 1 v: 1 c: 1)
  Down-pending: Disabled (e: -1 v: 1 c: 1)
  EOR interval: 300 EOR timer: Disabled (e: -1 v: 300)

  Force-Send interval: 2 Force-Send timer: Disabled (e: -1 v: 2)

  Rcv cap: Identity MP GR Refresh Security Overlay Network-Identity VSmart-Configuration
  Neg cap: Identity MP GR Refresh Security Overlay Network-Identity VSmart-Configuration
  Rcv afi-safi: TLOC-IPV4 SRVC-IPV4 SRVC-IPV6 ROUTE-IPV4 ROUTE-IPV6 MCAST-IPV4 (2) LINK CXP (2) IDENTITY-USER IDENTITY-IPV4-TO-USER
  IDENTITY-IPV6-TO-USER IDENTITY-IPV4-TO-SGT IDENTITY-IPV6-TO-SGT
  Neg afi-safi: TLOC-IPV4 SRVC-IPV4 SRVC-IPV6 ROUTE-IPV4 ROUTE-IPV6 MCAST-IPV4 (2) LINK CXP (2) IDENTITY-USER IDENTITY-IPV4-TO-USER
  IDENTITY-IPV6-TO-USER IDENTITY-IPV4-TO-SGT IDENTITY-IPV6-TO-SGT
  GR-enabled: Enabled, My GR interval: 43200 GR timer: Disabled (e: -1 v: 43200 c: 43200)
  Enter gr: 0, Exit gr: 0, GR mode: FALSE
  site-pol: SPOKE-SITES route-pol-in: None route-pol-out: CiscoLive-Active-Backup-Hub-Demo data-pol-in: None
  data-pol-out: None pfr-pol: None mem-pol: None cflowd:None
```

How can we verify if policy is being applied correctly?

```
Controller1#show omp peers 1.1.1.112 detail

peer                1.1.1.112
type                vedge
domain-id          1
site-id            112
overlay-id         1
region-id          None
state              up
version            1
legit              yes
control-up         yes
staging            no
upcount            4
downcount          3
last-uptime        2023-12-21T15:20:25+00:00
last-downtime      2023-12-21T15:20:23+00:00
uptime             17:20:45:00
hold-time          300
site-policy       SPOKE-SITES
policy-out      CiscoLive-Active-Backup-Hub-Demo
graceful-restart   supported
graceful-restart-interval 43200
network-identity   supported
identity-gr        not-supported
identity-gr-interval
refresh            supported
hello-sent         15520
hello-received     15542
handshake-sent     4
handshake-received 4
alert-sent         3
alert-received     0
inform-sent        42
inform-received    42
update-sent        1680
update-received    162
policy-sent
policy-received
```

Control policy applied.

Only incremented when centralized Data policy applied.

How to read OMP summary output?

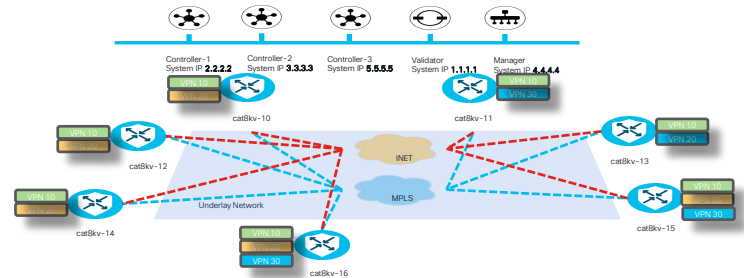
```

cat8kv-10#show sdwan omp summary
oper-state          UP
admin-state         UP
personality         vedge
device-role         Edge-Router
omp-uptime          0:00:34:34
routes-received     320
routes-installed    100
routes-sent         60
tlocs-received      38
tlocs-installed     12
tlocs-sent          6
services-received   4
services-installed  0
services-sent       12
mcast-routes-received 0
mcast-routes-installed 0
mcast-routes-sent   0
hello-sent          78
hello-received      63
handshake-sent      3
handshake-received  3
alert-sent          0
alert-received      0
inform-sent         39
inform-received     39
update-sent         63
update-received     255
policy-sent         0
policy-received     12
total-packets-sent  183
total-packets-received 375
vsmart-peers        3
    
```

Hello-sent	78	+
Handshake-sent	3	+
Alert-sent.	0	+
Inform-sent.	39	+
Update-sent	63	+
Policy-sent.	0	= 183 packets.

We have total 7 WAN-Edges with 2 TLOCs each, so total $6 \times 2 = 12$ TLOCs installed.

We have 2 TLOCs (private1, internet) as we have 3 OMP peering so total $3 \times 2 = 6$ TLOC-send.



We have 3 OMP peering and our policy is send in 4 fragments from 3 SD-WAN controllers. So, total policy packet received is 12.

How can we verify if policy is being applied correctly?

```
Controller1#show support policy route-policy
```

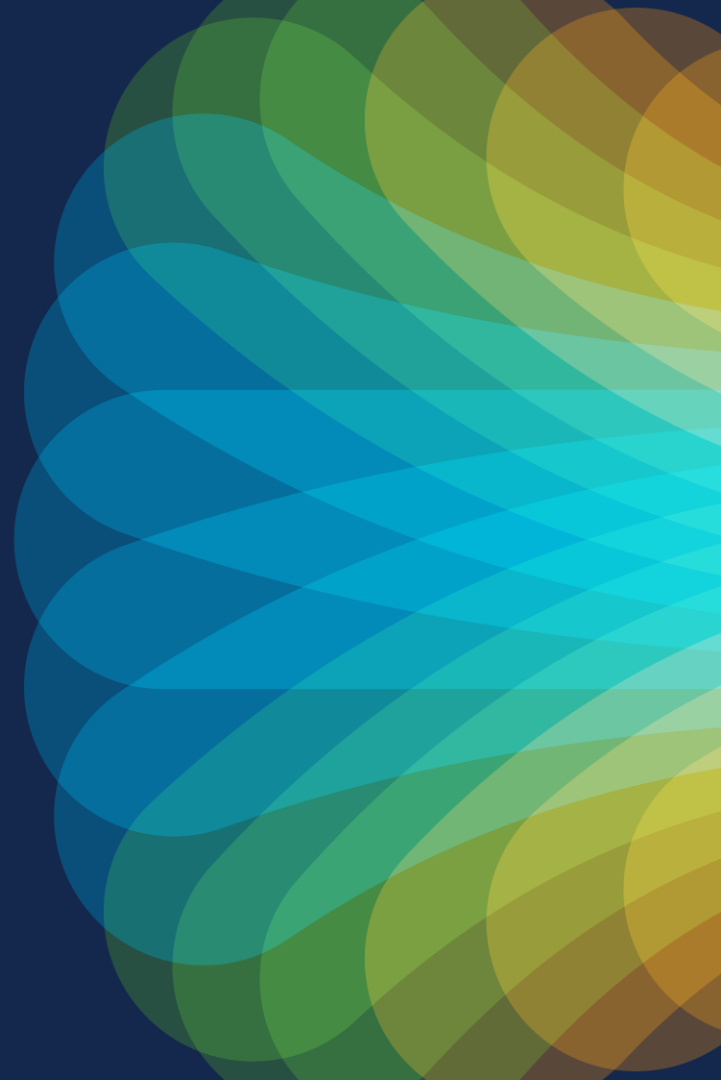
```
=====
ROUTE POLICIES
=====

route-policy CiscoLive-Active-Backup-Hub-Demo
seq-num 187
users-count 5
action srvc/srvc-chain/tloc/tloc-list/affinity counts: 0/0/0/1/0
Policy TLOC-Interest Database:
  TLOC:1.1.1.110 : mpls : ipsec Ref-Count: 1
  TLOC:1.1.1.110 : privatel : ipsec Ref-Count: 1
  TLOC:1.1.1.111 : mpls : ipsec Ref-Count: 1
  TLOC:1.1.1.111 : privatel : ipsec Ref-Count: 1

sequence: 1
  match tloc [SITE-LIST (0x1) ]
    site-list: SPOKE-SITES (0x7fd8ccfb3680)
  action: reject
  set: [ (0x0) ]
sequence: 11
  match tloc [SITE-LIST (0x1) ]
    site-list: DC-LIST (0x7fd94962af00)
  action: accept
  set: [ (0x0) ]
sequence: 21
  match route [SITE-LIST PFX-LIST (0x11) ]
    site-list: SPOKE-SITES (0x7fd8ccfb3680)
    IPv4 prefix-list: _AnyIpv4PrefixList (0x7fd8cd00a480)
  action: accept
  set: [TLOC-LIST (0x20) ]
    tloc-list: HUB-TLOCS [none]
  default-action: reject, fetch_xml: 1

Users:
1.1.1.112, type: route, dir: out, policy: CiscoLive-Active-Backup-Hub-Demo (0x7fd8cd01ae00), ctx: 0x7fd8cd031800, cb: 0x55fbf77bc4ac, change: no
1.1.1.113, type: route, dir: out, policy: CiscoLive-Active-Backup-Hub-Demo (0x7fd8cd01ae00), ctx: 0x7fd8cd034000, cb: 0x55fbf77bc4ac, change: no
1.1.1.114, type: route, dir: out, policy: CiscoLive-Active-Backup-Hub-Demo (0x7fd8cd01ae00), ctx: 0x7fd9493a0800, cb: 0x55fbf77bc4ac, change: no
1.1.1.115, type: route, dir: out, policy: CiscoLive-Active-Backup-Hub-Demo (0x7fd8cd01ae00), ctx: 0x7fd9496a7800, cb: 0x55fbf77bc4ac, change: no
1.1.1.116, type: route, dir: out, policy: CiscoLive-Active-Backup-Hub-Demo (0x7fd8cd01ae00), ctx: 0x7fd94939e000, cb: 0x55fbf77bc4ac, change: no
```

My OMP Peering is
down, where to start?



What steps should be taken if OMP peering fails to establish?



- Check if Control Connection (CC) is up between SD-WAN Edge and SD-WAN controller?
- If NOT, then we need to troubleshoot first, why CC is not coming up?



- If CC is UP but still OMP peering is not coming UP, then verify that **system-ip** is NOT overlapping in SD-WAN overlay.



- OMPd process crashed that cause OMP peering failed to establish.
- We can check the status if OMPd process crash/rebooted with the following commands
 - On SD-WAN Controller:
 - `show system status`
 - `show support omp peer peer-ip <PEER-IP>`
 - On SD-WAN Router:
 - `show platform software sdwan RP active internal "omp peer peer-ip <PEER-IP>"`
 - `show sdwan crash`
- Memory consumption/leakage can also cause OMP peering issue and can be seen with the following commands:
 - `[SD-WAN Edge] show platform software process memory r0 all sorted | include ompd`
 - `[SD-WAN Controller] show support omp memory-statistics`

Interesting SHOW commands for OMP debugging !!!



- SD-WAN Controller
 - `show omp peers`
 - `show omp peers <PEER-IP> detail`
 - `show omp summary`
 - `show support omp context`
 - `show support omp daemon`
 - `show support policy route-policy`
 - `show support omp peer peer-ip <PEER-IP>`
 - `show support omp rib vroute <VPN-ID:Prefix/Length>`
 - `show support omp memory-statistics`
 - `show support omp label-db`
- SD-WAN Edge
 - `show sdwan omp summary`
 - `show sdwan omp peers`
 - `show platform software sdwan RP active internal "omp daemon"`
 - `show platform software sdwan RP active internal "omp peer"`
 - `show platform software sdwan RP active internal "omp peer peer-ip <PEER-IP>"`
 - `show platform software sdwan RP active internal "omp rib vroute <VPN-ID:Prefix/Length>"`
 - `show platform software sdwan RP active internal "omp rib tloc <TLOC-IP:Color:Encap>"`
 - `show platform software sdwan RP active internal "omp memory-statistics"`

What is OMP Graceful Restart?

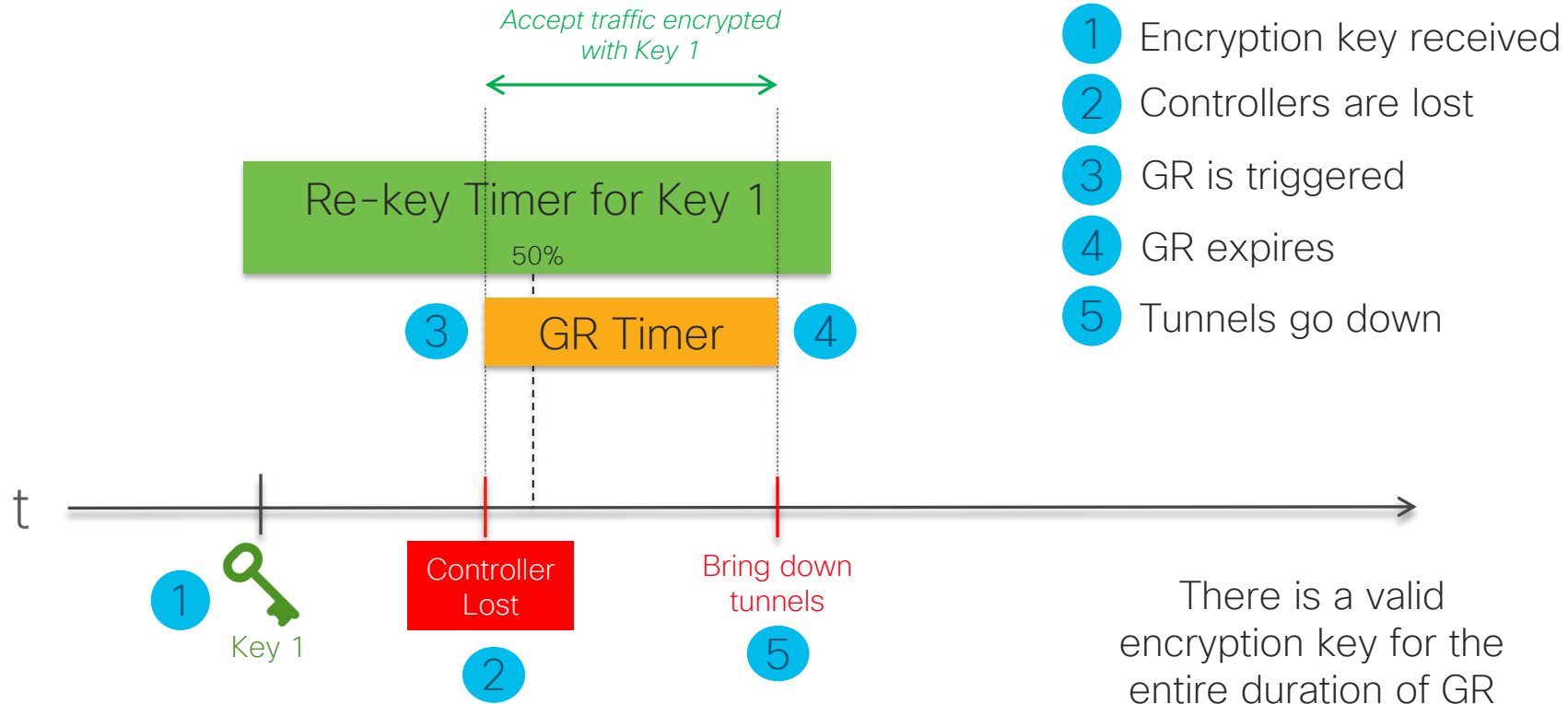
OMP graceful restart allows OMP peers to continue operating if one of the peers becomes unavailable.

WAN Edge router and a SD-WAN controller cache the OMP information that they learn from their peer.

OMP graceful restart timer tells the OMP peer **how long to retain** the cached advertised routes.

Loss During First 50% of IPsec Re-Key Timer

Re-key timer twice or more OMP Graceful Restart timer



OMP Graceful Restart – Off

Device Templates Feature Templates

Feature Template > OMP > controller-omp

Device Type vSmart

Template Name* controller-omp

Description* controller-omp

Basic Configuration Timers

▼ BASIC CONFIGURATION

Graceful Restart for OMP On Off

Graceful Restart Timer (seconds) 43200

Number of Paths Advertised per Prefix [omp_send_path_limit]

Send Backup Paths On Off

- Graceful Restart for OMP = **Off** at all SD-WAN Controllers means **Graceful Restart is disabled** on **all SD-WAN Edges**.
- OMP Routing Table and BFD Session are lost immediately when SD-WAN Edges loses **all OMP peers**.

```
Controller1# show running-config omp
omp
no shutdown
filter-route
  no outbound affinity-group-preference
  no outbound tloc-color
exit
no graceful-restart
outbound-policy-caching
!
```

```
Controller2# show running-config omp
omp
no shutdown
filter-route
  no outbound affinity-group-preference
  no outbound tloc-color
exit
no graceful-restart
outbound-policy-caching
!
```

```
cat8kv-10#show sdwan omp peers detail | include graceful|peer|state|type
peer                2.2.2.2
type                vsmart
state              up
graceful-restart   not-supported
graceful-restart-interval
peer                3.3.3.3
type                vsmart
state              up
graceful-restart   not-supported
graceful-restart-interval
peer                5.5.5.5
type                vsmart
state              up
graceful-restart   not-supported
graceful-restart-interval
```

```
cat8kv-10# !! -- #### shutdown Controller1, Controller2 and Controller3 #### -- !!
Site400-cE1#
2023/12/21 11:42:03.988025744 {iosrp_R0-0}{255}: [iosrp] [17261]: (info): *Dec 21
11:42:03.988: %Cisco-SDWAN-cat8kv-10-OMPD-6-INFO-1400002: Notification: 12/21/2023
11:42:3 omp-number-of-vsmarts-change severity-level:major host-name:"cat8kv-10"
system-ip:1.1.1.110 tenant-name:"[Default]" tenant-global-id:0 number-of-vsmarts:0
cat8kv-10#show sdwan bfd sessions
cat8kv-10#
```

OMP Graceful Restart Timer

Device Templates Feature Templates

Feature Template > OMP > controller-omp

Device Type vSmart

Template Name* controller-omp

Description* controller-omp

Basic Configuration Timers

BASIC CONFIGURATION

Graceful Restart for OMP On Off

Graceful Restart Timer (seconds) 86400

Number of Paths Advertised per Prefix 4

Send Backup Paths On Off

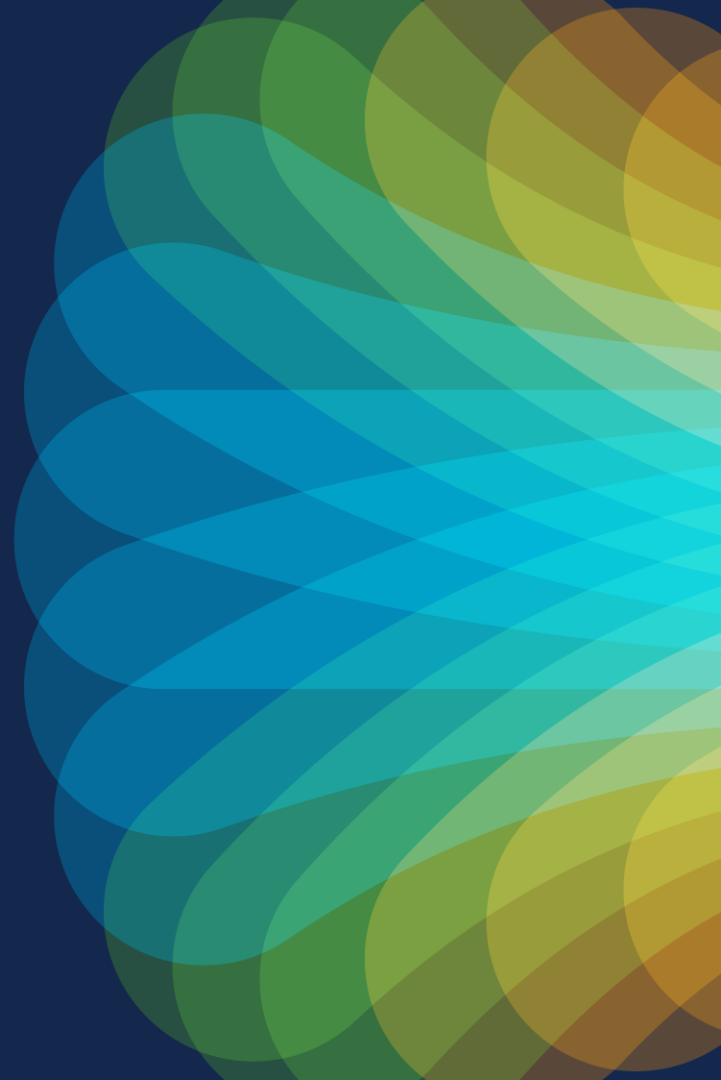
- Graceful Restart timer configured on SD-WAN controllers is applied to SD-WAN Edge, and conversely.
- If any change to an **OMP graceful restart configuration** is made, the OMP session between the Cisco SD-WAN controllers and the device is flapped.

```
Controller1# show running-config omp
omp
no shutdown
filter-route
  no outbound affinity-group-preference
  no outbound tloc-color
exit
graceful-restart
outbound-policy-caching
timers
  graceful-restart-timer 86400
exit
!
```

```
Controller1#show omp peers 1.1.1.110 detail | include peer\|state\|graceful
peer      1.1.1.110
state     up
graceful-restart supported
graceful-restart-interval 43200
```

```
cat8kv-10#show sdwan omp peers detail | include peer\|state\|graceful
peer      2.2.2.2
state     up
graceful-restart supported
graceful-restart-interval 86400
peer      3.3.3.3
state     up
graceful-restart supported
graceful-restart-interval 86400
```

SD-WAN Controller RIB-IN/RIB-OUT Calculation



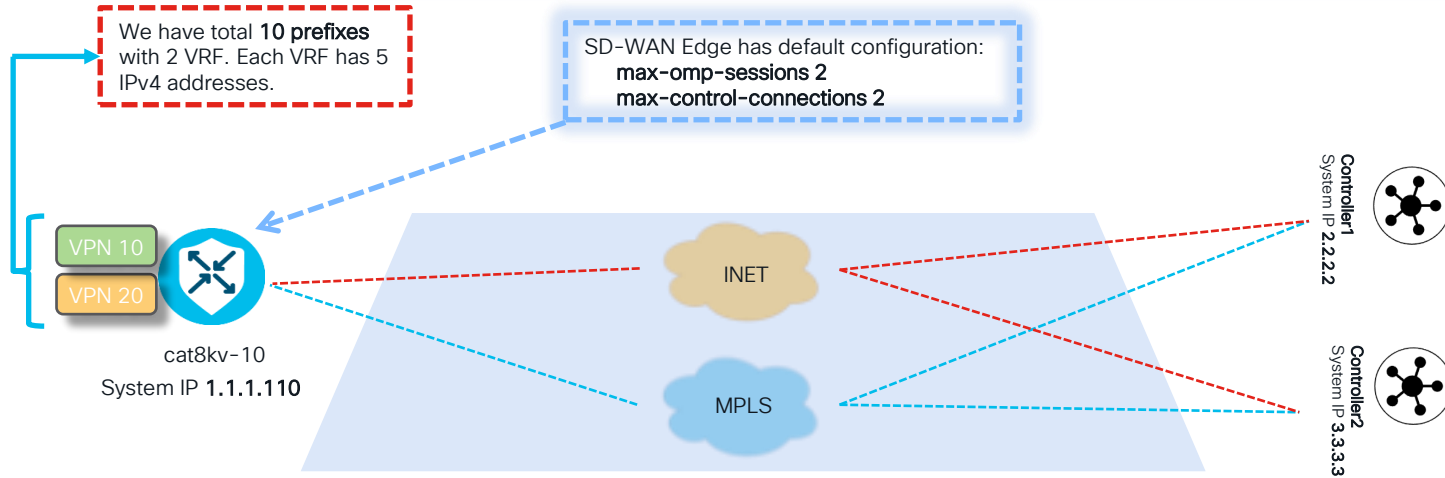
SD-WAN Controller RIB-IN calculation

SD-WAN Controller Peer Received routes (RIB-in) calculation = WAN Edge received routes +

(n x WAN Edge received routes per SD-WAN controller)

n = Number of SD-WAN Controller peers with edge devices

WAN-Edge received routes = Number of routers with OMP sessions (directly connected) to the SD-WAN Controller X
Number of prefixes sent per router X
Number of TLOCs per router



SD-WAN Controller RIB-IN calculation

SD-WAN **Controller1** per Peer (**1.1.1.110**) received IPv4 routes (RIB-in) calculation = WAN Edge IPv4 received routes +
(n x WAN Edge IPv4 received routes per SD-WAN controller)

n = Number of SD-WAN Controller peers with edge devices

SD-WAN **Controller1** per Peer (**1.1.1.110**) received IPv4 routes (RIB-in) calculation = **20 + (1 X 20) = 40**

WAN-Edge IPv4 received routes = Number of routers with OMP sessions (directly connected) to the SD-WAN **Controller1** (**1**) X
Number of prefixes sent per router (**10**) X
Number of TLOCs per router (**2**) = **1 X 10 X 2 = 20**

We have total **10 prefixes**
with 2 VRF. Each VRF has 5
IPv4 addresses.

cat8kv-10
System IP 1.1.1.110

VPN 10
VPN 20

Controller1
System IP 2.2.2.2

Controller2
System IP 3.3.3.3

SD-WAN Edge has default configuration:
max-omp-sessions 2
max-control-connections 2

↔ OMP Peering

SD-WAN Controller RIB-IN calculation

SD-WAN **Controller1** per Peer (1.1.1.110) received IPv4 routes (RIB-in) calculation = WAN Edge IPv4 received routes +

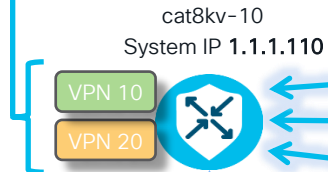
(n x WAN Edge IPv4 received routes per SD-WAN controller)

n = Number of SD-WAN Controller peers with edge devices

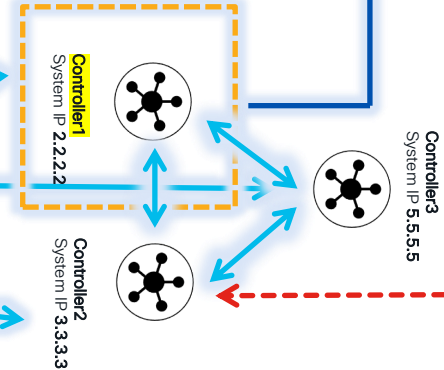
SD-WAN **Controller1** per Peer (1.1.1.110) received IPv4 routes (RIB-in) calculation = 20 + (2 X 20) = 20 + 40 = **60**

WAN-Edge IPv4 received routes = Number of routers with OMP sessions (directly connected) to the SD-WAN **Controller1** (1) X
 Number of prefixes sent per router (10) X
 Number of TLOCs per router (2) = 1 X 10 X 2 = **20**

We have total **10 prefixes** with 2 VRF. Each VRF has 5 IPv4 addresses.



SD-WAN Edge has configuration:
max-omp-sessions 3
max-control-connections 3



↔ OMP Peering

SD-WAN Controller RIB-OUT calculation – Full Mesh Topology

Send routes (RIB-OUT) calculation = (WAN-Edge received routes X Number of Controller Peers) +
(WAN-Edge received routes X (Number of Router Peers - 1)) +
(Controller Received Routes X Number of Router Peers)

Where

WAN-Edge received routes = Number of routers with OMP sessions (directly connected) to the SD-WAN Controller X
Number of prefixes sent per router X
Number of TLOCs per router

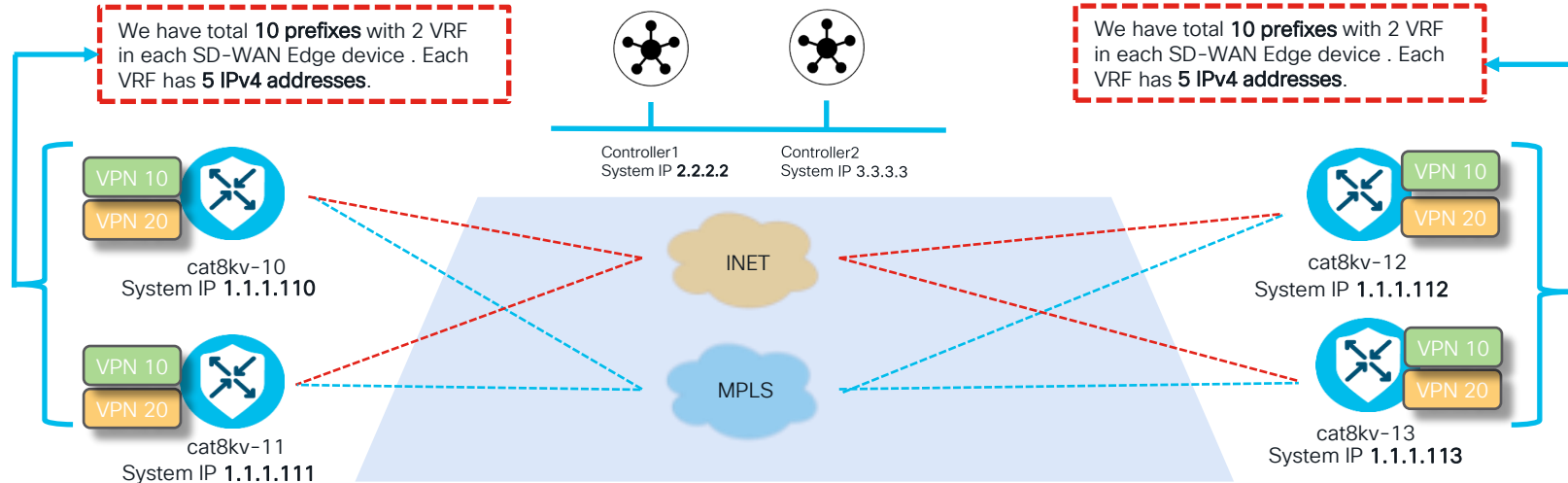
Controller Received Routes = (Number of Routers not directly connected) X (Prefixes per Router) X (Number of TLOC's)

SD-WAN Edge has default configuration:

max-omp-sessions 2
max-control-connections 2

We have total **10 prefixes** with 2 VRF in each SD-WAN Edge device . Each VRF has **5 IPv4 addresses**.

We have total **10 prefixes** with 2 VRF in each SD-WAN Edge device . Each VRF has **5 IPv4 addresses**.



SD-WAN Controller RIB-OUT calculation – Full Mesh Topology

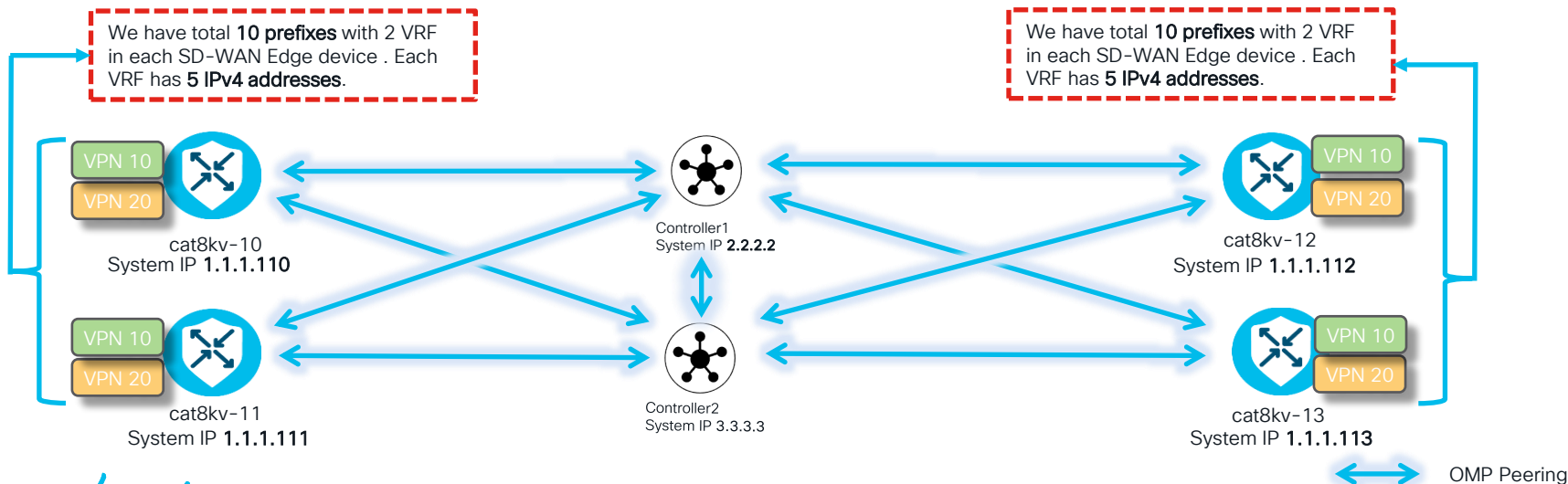
SD-WAN Edge has default configuration:

max-omp-sessions 2
max-control-connections 2

Send routes (RIB-OUT) calculation = (WAN-Edge received routes X Number of Controller Peers) +
(WAN-Edge received routes X (Number of Router Peers - 1)) +
(Controller Received Routes X Number of Router Peers)

WAN-Edge received routes = Number of routers with OMP sessions (directly connected) to the SD-WAN Controller X
Number of prefixes sent per router X
Number of TLOCs per router

Controller Received Routes = (Number of Routers not directly connected) X (Prefixes per Router) X (Number of TLOC's)



SD-WAN Controller RIB-OUT calculation – Full Mesh Topology

SD-WAN Edge has default configuration:
max-omp-sessions 2
max-control-connections 2

Send routes (RIB-OUT) calculation = (WAN-Edge received routes X Number of SD-WAN Controller peers to SD-WAN **Controller1**) +
 (WAN-Edge received routes X (Number of SD-WAN Edge router peers to SD-WAN **Controller1** - 1)) +
 (Controller Received Routes X Number of SD-WAN Edge router peers to SD-WAN **Controller1**)

Send routes (RIB-OUT) calculation = (80 X 1) + (80 X (4 - 1)) + (0 X 4) = 80 + (80 X 3) + 0 = 320

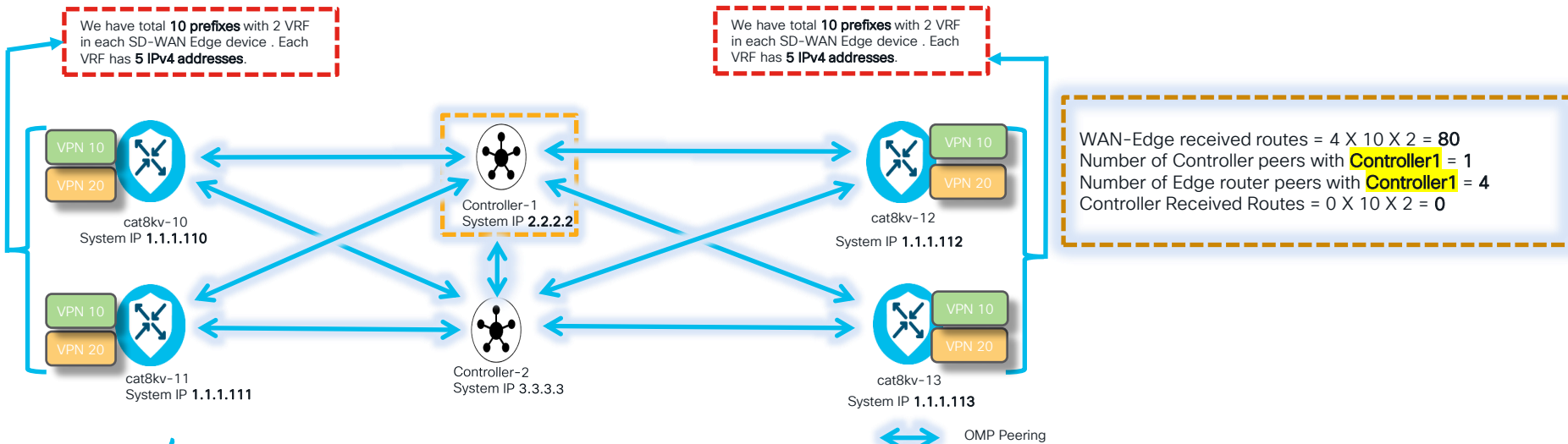
WAN-Edge received routes = Number of routers with OMP sessions (directly connected) to the SD-WAN **Controller1** X
 Number of prefixes sent per router X
 Number of TLOCs per router

Controller Received Routes = (Number of Routers not directly peered) X (Prefixes per Router) X (Number of TLOC's)

We have total **10 prefixes** with 2 VRF in each SD-WAN Edge device . Each VRF has **5 IPv4 addresses**.

We have total **10 prefixes** with 2 VRF in each SD-WAN Edge device . Each VRF has **5 IPv4 addresses**.

WAN-Edge received routes = 4 X 10 X 2 = **80**
 Number of Controller peers with **Controller1** = 1
 Number of Edge router peers with **Controller1** = 4
 Controller Received Routes = 0 X 10 X 2 = **0**



SD-WAN Controller RIB-OUT calculation – Full Mesh Topology

SD-WAN Edge has default configuration:
max-omp-sessions 2
max-control-connections 2

Send routes (RIB-OUT) calculation = (WAN-Edge received routes X Number of SD-WAN Controller peers to SD-WAN **Controller1**) + (WAN-Edge received routes X (Number of SD-WAN Edge router peers to SD-WAN **Controller1** - 1)) + (Controller Received Routes X Number of SD-WAN Edge router peers to SD-WAN **Controller1**)

Send routes (RIB-OUT) calculation = (40 X 3) + (40 X (2 - 1)) + (40 X 2) = 120 + (40 X 1) + 80 = **240**

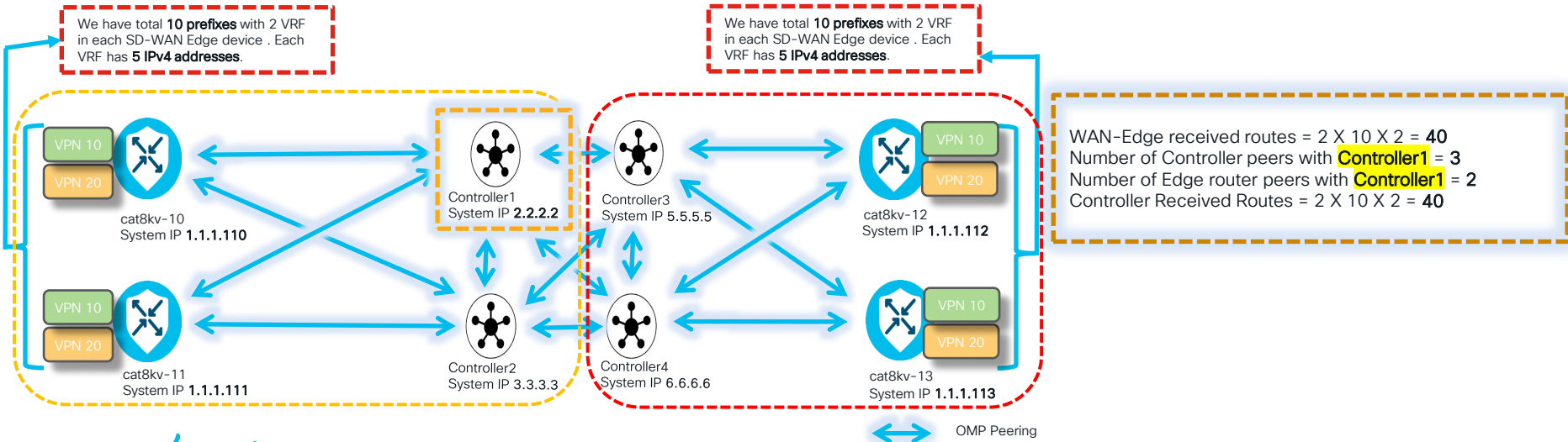
WAN-Edge received routes = Number of routers with OMP sessions (directly connected) to the SD-WAN **Controller-1** X Number of prefixes sent per router X Number of TLOCs per router

Controller Received Routes = (Number of Routers not directly connected) X (Prefixes per Router) X (Number of TLOC's)

We have total **10 prefixes** with 2 VRF in each SD-WAN Edge device . Each VRF has **5 IPv4 addresses**.

We have total **10 prefixes** with 2 VRF in each SD-WAN Edge device . Each VRF has **5 IPv4 addresses**.

WAN-Edge received routes = 2 X 10 X 2 = **40**
 Number of Controller peers with **Controller1** = **3**
 Number of Edge router peers with **Controller1** = **2**
 Controller Received Routes = 2 X 10 X 2 = **40**



HUB and Spoke Topology

- Hub-and-spoke topologies are typically implemented using one of the following methods:
 - Use of centralized control policy to rewrite the TLOC.
 - Use of centralized control policy to filter out the spoke prefixes.
- **Use of centralized control policy to rewrite the TLOC:**
 - TLOC of the spoke router is replaced with one or more TLOCs which represent the hub routers.
 - TLOC rewrite policy is generally deployed as an outbound centralized control policy.
- **Use of centralized control policy to filter out the spoke prefixes:**
 - Spoke prefixes/routes are being filtered by SD-WAN controller to each spoke router.
 - ONLY default route (optionally hub prefixes/routes) is sent and reachable via TLOC of the hubs.

SD-WAN Controller RIB-OUT calculation – Hub and Spoke TLOC re-write policy 1/2



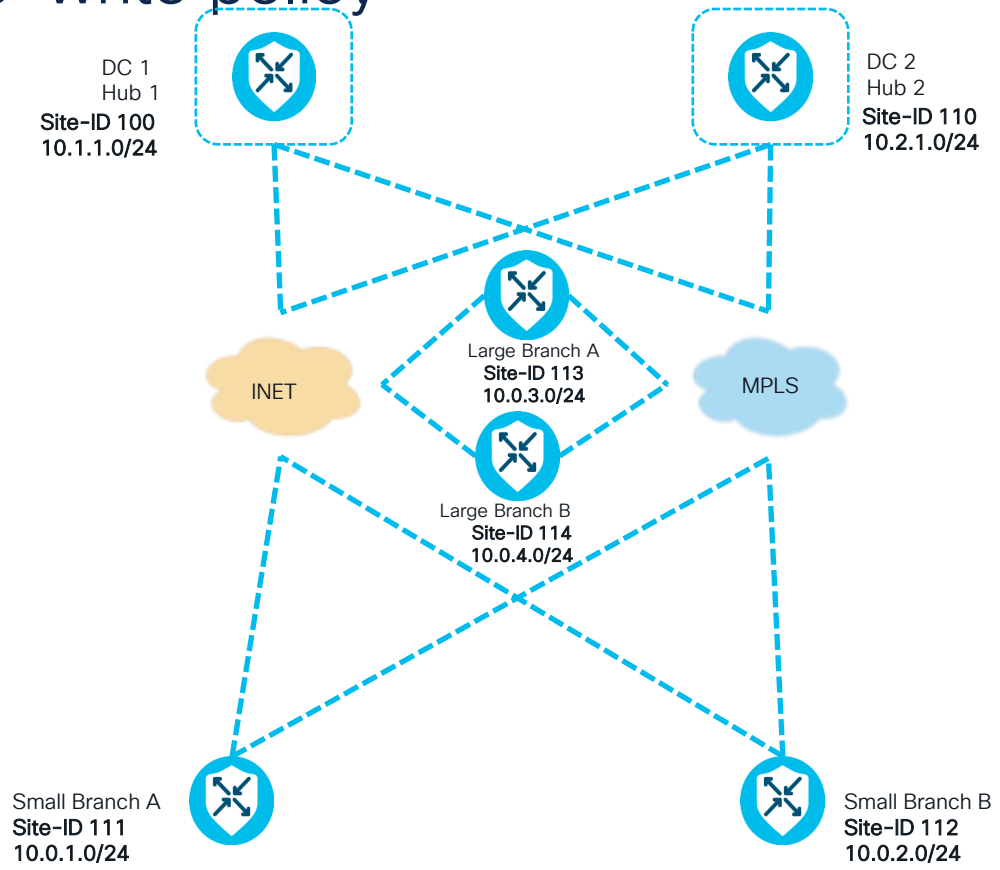
- I. In Hub and Spoke topology, all SD-WAN Edge routers (hub or spoke) have OMP peered with the SD-WAN controller, and which are then reflected to other SD-WAN controllers within overlay as all SD-WAN controllers have OMP peering with each other.
 - SD-WAN Edge received routes \times Number of SD-WAN controllers peers with the current SD-WAN controller
- II. SD-WAN Edge routes received from the spoke that are OMP-peered with the SD-WAN controller which are then reflected to other SD-WAN Edge spoke routers that have OMP peering with the same SD-WAN controller. As SD-WAN Edge routes is replaced with the one or more TLOCs of the hub routers.
 - SD-WAN Edge spoke received routes \times Number of HUB TLOCs \times (Number of SD-WAN Edge spoke routers peered with the current SD-WAN controller - 1)
- III. SD-WAN Edge routes from spoke routers that are OMP peered with the SD-WAN controller, which are then reflected to HUB router that are OMP peered with the current SD-WAN controller. In this case No TLOC rewrite was done as spoke routes are advertised to HUB routers.
 - SD-WAN Edge spoke received routes \times Number of SD-WAN Edge HUB routers that have OMP peered with the current SD-WAN controller
- IV. SD-WAN Edge routes from hub routers that are OMP peered with the SD-WAN controller, which are then reflected to spoke router that are OMP peered with the current SD-WAN controller. In this case No TLOC rewrite was not apply to HUB routers.
 - SD-WAN Edge hub received routes \times Number of SD-WAN Edge spoke routers that have OMP peered with the current SD-WAN controller

SD-WAN Controller RIB-OUT calculation – Hub and Spoke TLOC re-write policy 2/2

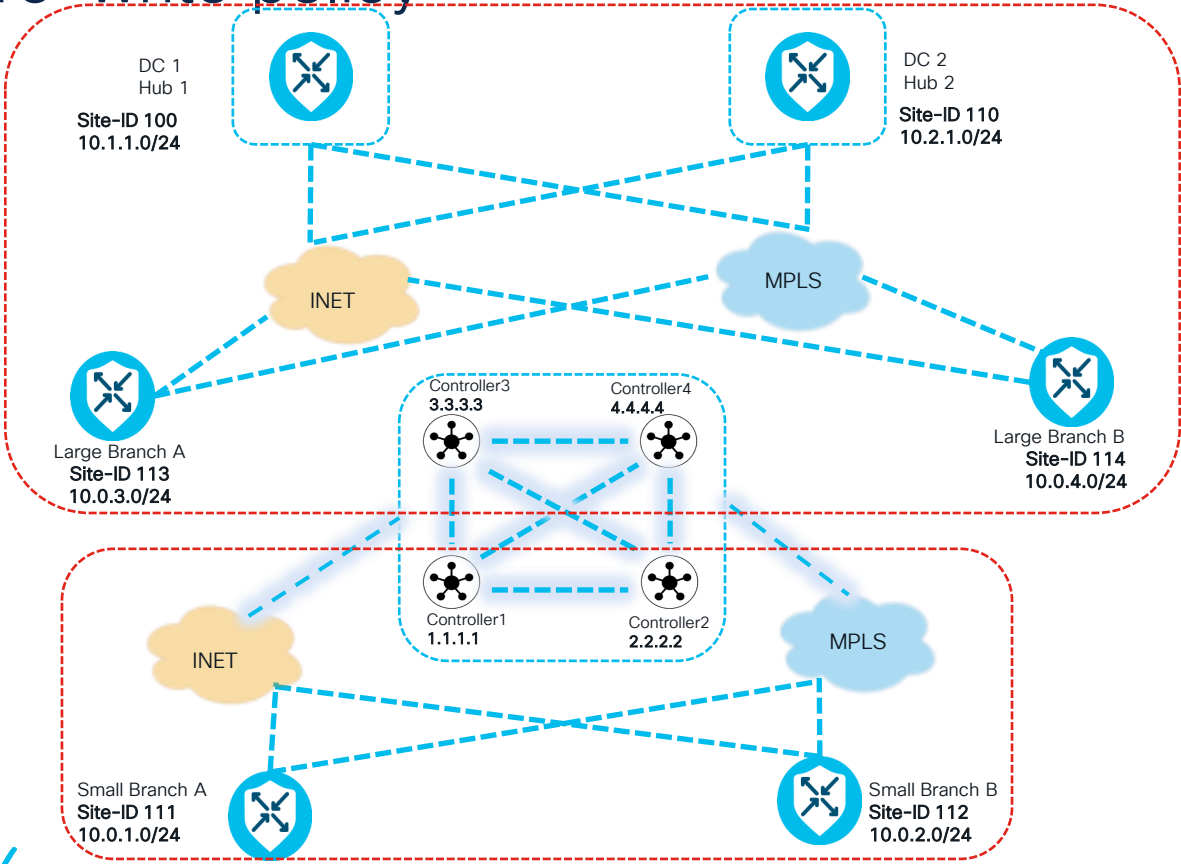


- V. Now we take into consideration the SD-WAN Edge spoke received routes from **other SD-WAN Controller** in the network which are reflected to the SD-WAN Controller and, in turn, reflected to the SD-WAN Edge spoke routers OMP-peered with the current SD-WAN Controller. Now SD-WAN Edge spoke routes are replaced with the TLOC's of Hub routers.
- Number of SD-WAN spoke received routes from other SD-WAN Controller instances which will be reflected multiplied by the number of TLOCs per device X Number of HUB TLOC's through which the spoke edge received routes will be advertised X Number of SD-WAN spoke router peers to the current SD-WAN Controller
- vi. Next, we need to account for the SD-WAN Edge spoke received routes from other SD-WAN Controller in the network which are reflected to the SD-WAN Controller and, in turn, reflected to the HUB SD-WAN routers OMP-peered with the SD-WAN Controller.
- Number of SD-WAN spoke received routes from other SD-WAN Controller X Number of HUB router peers to the given SD-WAN Controller
- VII. In the final part, we take into account the HUB edge received routes from other SD-WAN Controller in the network which are reflected to the SD-WAN Controller and, in turn, reflected to the both the hub and spoke SD-WAN Edge routers OMP-peered with the SD-WAN
- Number of hub edge received routes from other SD-WAN Controllers which will be reflected X Number of router (hub and spoke) peers to the given SD-WAN Controller

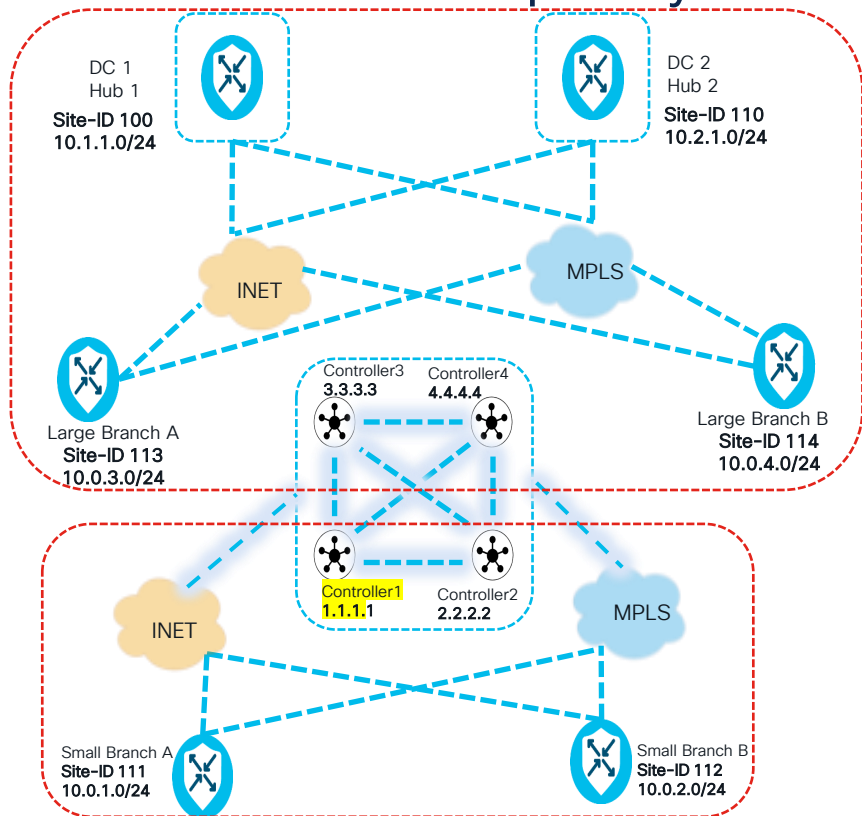
SD-WAN Controller RIB-OUT calculation – Hub and Spoke TLOC re-write policy



SD-WAN Controller RIB-OUT calculation – Hub and Spoke TLOC re-write policy

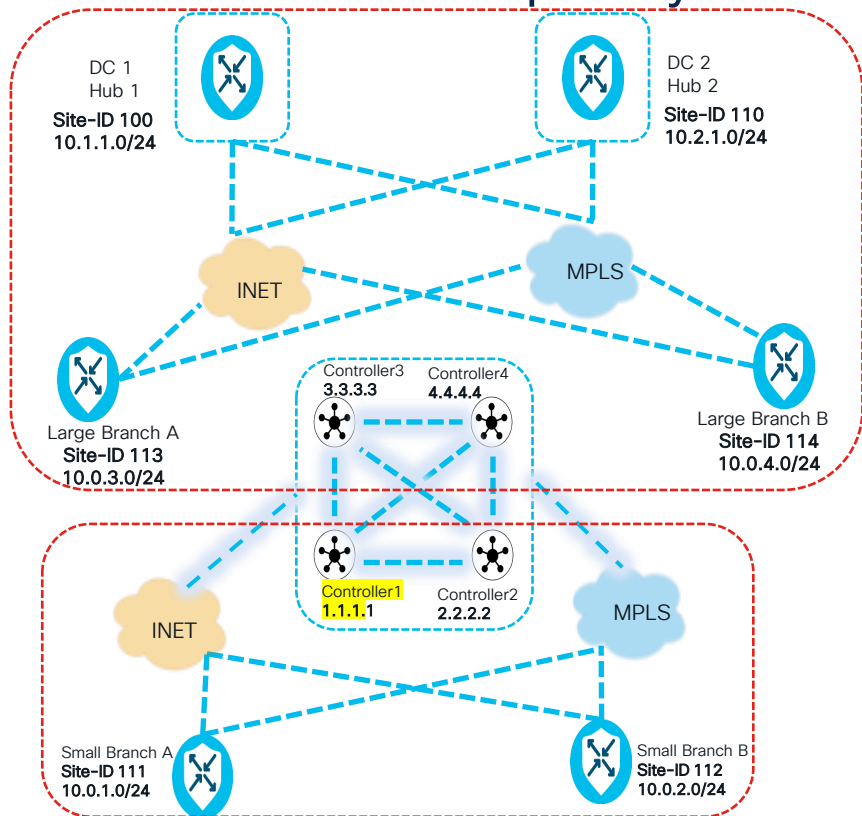


SD-WAN Controller RIB-OUT calculation – Hub and Spoke TLOC re-write policy



- I. First component of RIB-OUT for SD-WAN **Controller1**
 - SD-WAN Edge received routes **X** Number of SD-WAN controllers peers with the **SD-WAN Controller1** = $4 \times 3 = 12$
 - SD-WAN Edge received routes = Number of SD-WAN Edge routers OMP peered with **SD-WAN Controller1** **X** (Prefixes per SD-WAN Router) **X** (Number of TLOCs) = $2 \times 1 \times 2 = 4$
 - Number of SD-WAN controllers peers with the current **SD-WAN Controller-1** = 3
- II. Second component of RIB-OUT for SD-WAN **Controller1**
 - SD-WAN Edge spoke received routes **X** Number of HUB TLOCs **X** (Number of SD-WAN Edge spoke routers peered with the **SD-WAN Controller1** - 1) = $4 \times 4 (2 - 1) = 16$
 - SD-WAN Edge spoke received routes = Number of spokes routers OMP peered with SD-WAN **Controller-1** **X** (Prefixes per SD-WAN Router) **X** (Number of TLOCs) = $2 \times 1 \times 2 = 4$
 - Number of HUB TLOCs = 4
 - Number of SD-WAN Edge spoke routers peered with the current SD-WAN **Controller1** = 2

SD-WAN Controller RIB-OUT calculation – Hub and Spoke TLOC re-write policy



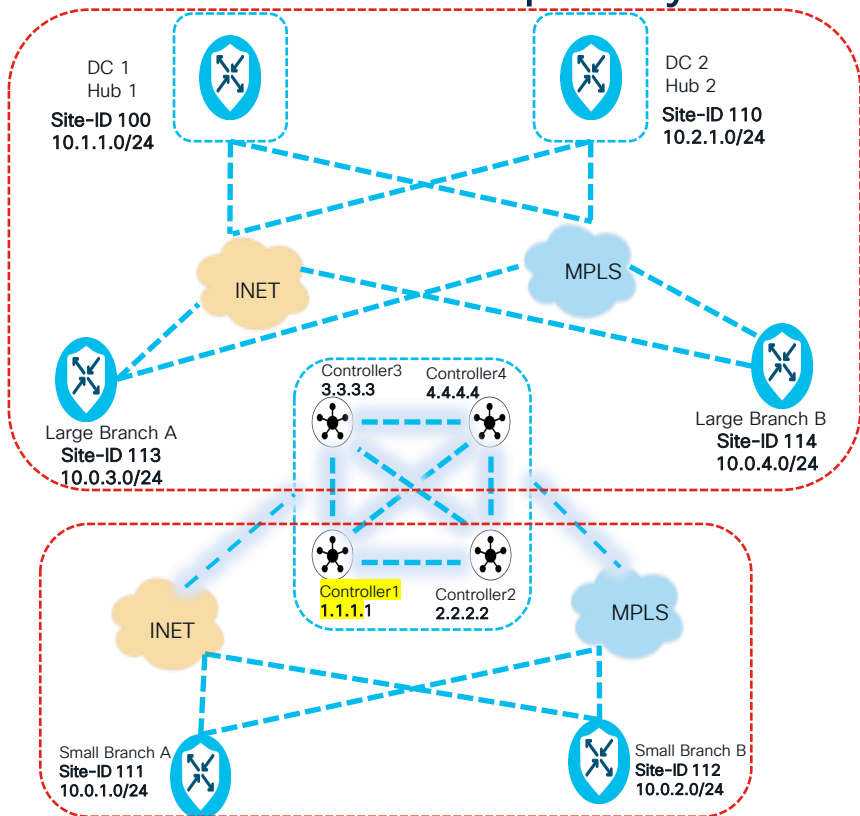
III. Third component of RIB-OUT for SD-WAN Controller1

- SD-WAN Edge spoke received routes X Number of SD-WAN Edge HUB routers that have OMP peered with the SD-WAN Controller1 = $4 \times 0 = 0$
 - SD-WAN Edge spoke received routes = Number of spokes routers OMP peered with SD-WAN Controller1 X (Prefixes per SD-WAN Router) X (Number of TLOCs) = $2 \times 1 \times 2 = 4$
 - Number of SD-WAN Edge HUB routers that have OMP peered with the SD-WAN Controller1 = 0

IV. Fourth component of RIB-OUT for SD-WAN Controller1

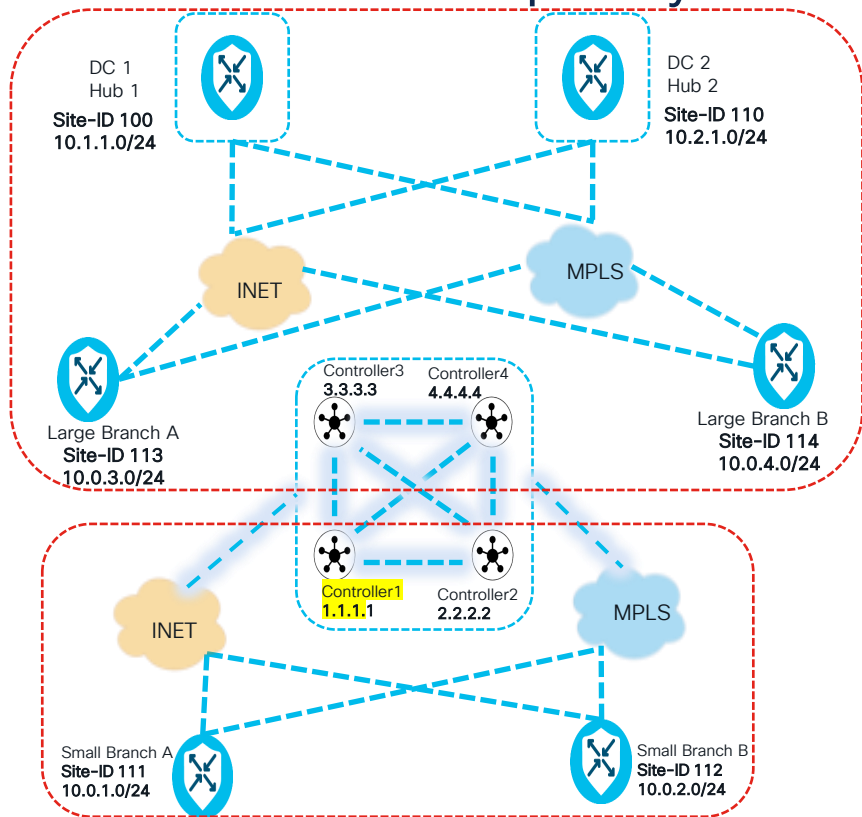
- SD-WAN Edge hub received routes X Number of SD-WAN Edge spoke routers that have OMP peered with the SD-WAN Controller1 = $0 \times 2 = 0$
 - SD-WAN Edge hub received routes = Number of HUB routers OMP peered with SD-WAN Controller1 X (Prefixes per SD-WAN Router) X (Number of TLOCs) = $0 \times 2 \times 2 = 0$
 - Number of SD-WAN Edge spoke routers that have OMP peered with the SD-WAN Controller1 = 2

SD-WAN Controller RIB-OUT calculation – Hub and Spoke TLOC re-write policy



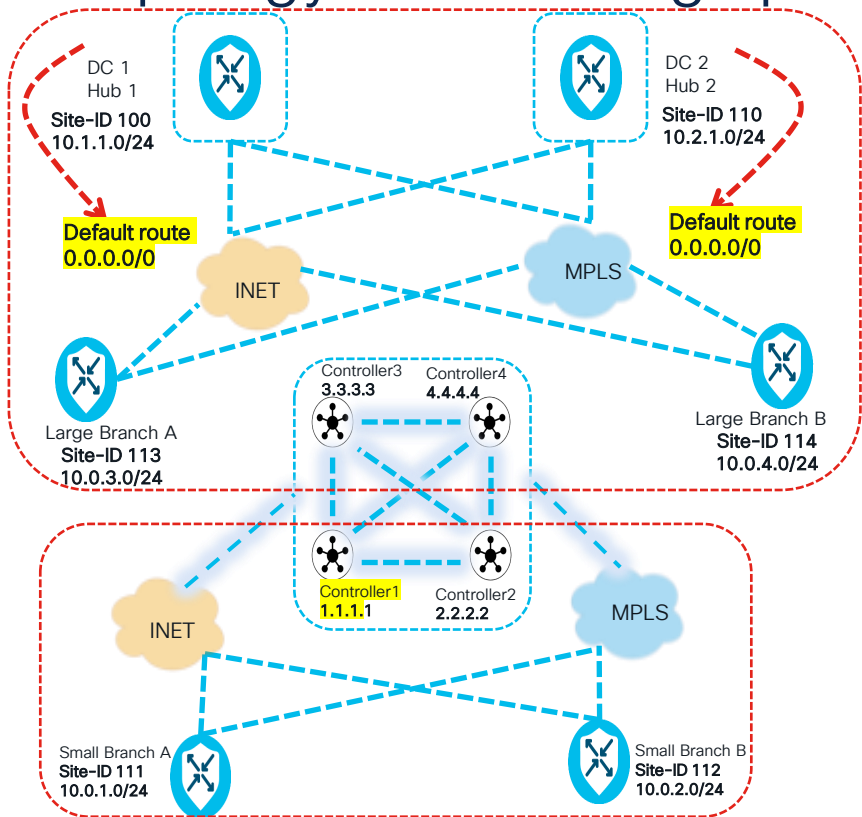
- V. Fifth component of RIB-OUT for SD-WAN **Controller1**
- Number of SD-WAN spoke received routes from other **SD-WAN Controller1** which will be reflected multiplied by the number of TLOCs per device **X** Number of HUB TLOC's through which the spoke edge received routes will be advertised **X** Number of SD-WAN spoke router peers to the **SD-WAN Controller1** = $4 \times 4 \times 2 = 32$
 - Number of SD-WAN spoke received routes from other **SD-WAN Controller** which will be reflected multiplied by the number of TLOCs per device = $(1 \text{ Prefix from Large Branch A} \times 2 \text{ TLOC}) + (1 \text{ Prefix from Large Branch B} \times 2 \text{ TLOC}) = 2 + 2 = 4$
 - Number of HUB TLOC's through which the spoke edge received routes will be advertised to be available = 4
 - Number of SD-WAN spoke router peers to the **SD-WAN Controller1** = 2
- VI. Sixth component of RIB-OUT for SD-WAN **Controller1**
- Number of SD-WAN spoke received routes from other **SD-WAN Controller X** Number of HUB router peers to the given SD-WAN Controller = $4 \times 0 = 0$
 - SD-WAN Edge spoke received routes = $(1 \text{ Prefix from Large Branch A} \times 2 \text{ TLOC}) + (1 \text{ Prefix from Large Branch B} \times 2 \text{ TLOC}) = 2 + 2 = 4$
 - Number of **SD-WAN Edge HUB** routers that have OMP peered with the **SD-WAN Controller1** = 0

SD-WAN Controller RIB-OUT calculation – Hub and Spoke TLOC re-write policy



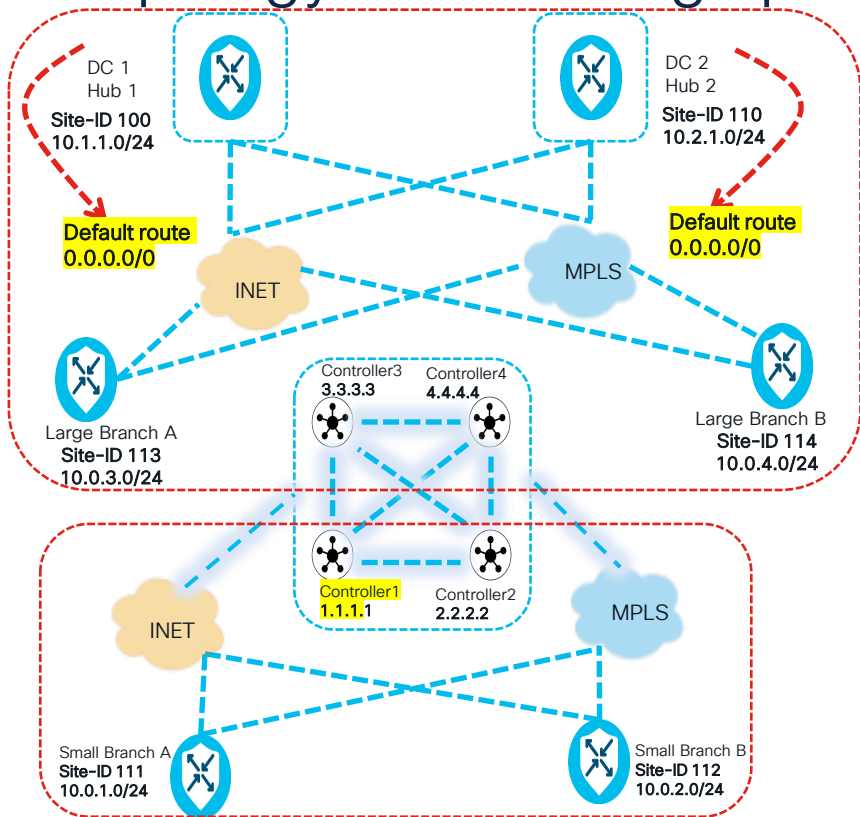
- V. Final component of RIB-OUT for SD-WAN **Controller1**
- Number of hub edge received routes from other SD-WAN Controllers which will be reflected \times Number of router (hub and spoke) peers to the given SD-WAN Controller = $4 \times 2 = 8$
 - Number of SD-WAN **HUB** received routes from other SD-WAN Controller which will be reflected multiplied by the number of TLOCs per device = $(1 \text{ Prefix from HUB 1} \times 2 \text{ TLOC}) + (1 \text{ Prefix from HUB 2} \times 2 \text{ TLOC}) = 2 + 2 = 4$
 - Number of router (either hub or spoke) peers to the given SD-WAN Controller = 2
 - Now Summing all seven components gives us the total routes sent for **Controller1 (RIB-OUT)** = $12 + 16 + 0 + 0 + 32 + 0 + 8 = 68$ routes

SD-WAN Controller RIB-OUT calculation – Hub and Spoke topology with filtering spoke routes



- I. First component of RIB-OUT for SD-WAN **Controller1**
 - SD-WAN Edge received routes **X** Number of SD-WAN controllers peers with the **SD-WAN Controller1** = $4 \times 3 = 12$
 - SD-WAN Edge received routes = Number of SD-WAN Edge routers OMP peered with **SD-WAN Controller1** **X** (Prefixes per SD-WAN Router) **X** (Number of TLOCS) = $2 \times 1 \times 2 = 4$
 - Number of SD-WAN controllers peers with the current **SD-WAN Controller1** = 3
- II. Second component of RIB-OUT for SD-WAN **Controller1**
 - SD-WAN Edge spoke received routes **X** Number of SD-WAN Edge HUB routers peered with the **SD-WAN Controller1** = $4 \times 0 = 0$
 - SD-WAN Edge spoke received routes = Number of spokes routers OMP peered with SD-WAN **Controller1** **X** (Prefixes per SD-WAN Router) **X** (Number of TLOCS) = $2 \times 1 \times 2 = 4$
 - Number of SD-WAN Edge **HUB** routers peered with the current SD-WAN **Controller1** = 0

SD-WAN Controller RIB-OUT calculation – Hub and Spoke topology with filtering spoke routes



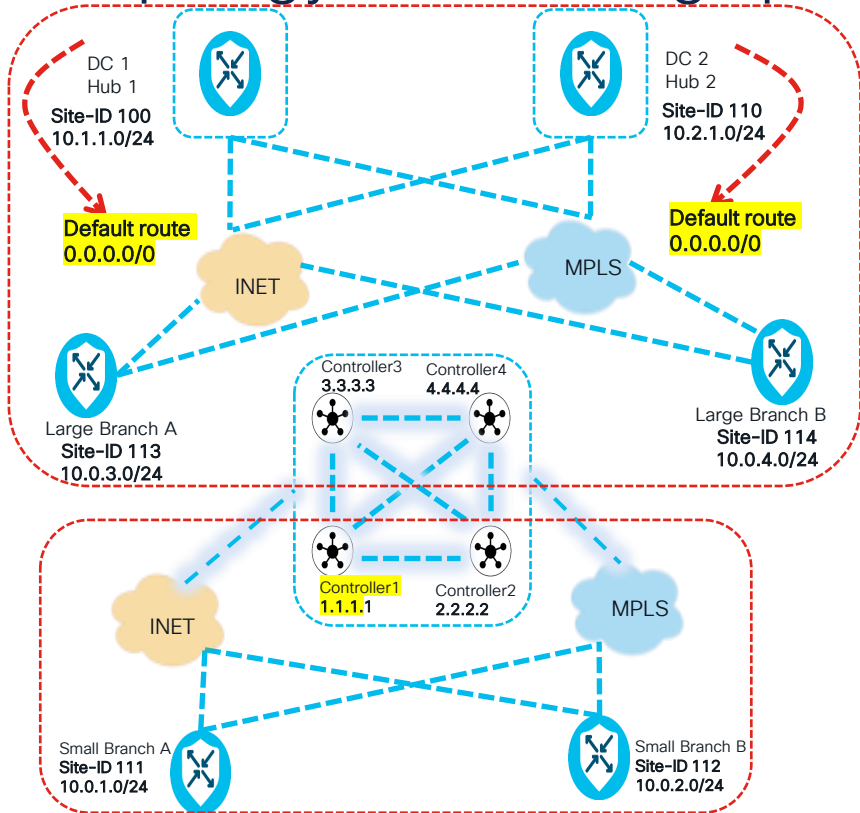
III. Third component of RIB-OUT for SD-WAN Controller1

- SD-WAN Edge HUB received routes X Number of SD-WAN Edge spoke routers that have OMP peered with the SD-WAN Controller1 = $0 \times 2 = 0$
 - SD-WAN Edge HUB received routes = Number of HUB routers OMP peered with SD-WAN Controller1 X (Prefixes per SD-WAN Router) X (Number of TLOCs) = $(0 \times 1 \times 2) = 0$
 - Number of SD-WAN Edge spoke routers that have OMP peered with the SD-WAN Controller1 = 2

IV. Fourth component of RIB-OUT for SD-WAN Controller1

- Number of SD-WAN Edge spoke received routes from other SD-WAN controllers X Number of SD-WAN Edge HUB routers that have OMP peered with the SD-WAN Controller1 = $4 \times 0 = 0$
 - Number of SD-WAN Edge spoke received routes from other SD-WAN controllers = $(1 \text{ Prefix from Large Branch A} \times 2 \text{ TLOC}) + (1 \text{ Prefix from Large Branch B} \times 2 \text{ TLOC}) = 2 + 2 = 4$
 - Number of SD-WAN Edge HUB routers that have OMP peered with the SD-WAN Controller1 = 0

SD-WAN Controller RIB-OUT calculation – Hub and Spoke topology with filtering spoke routes



- V. Final component of RIB-OUT for SD-WAN **Controller1**
- Number of SD-WAN **HUB** received **(INCLUDING the DEFAULT)** from other SD-WAN Controllers **X** Number of SD-WAN Edge **spoke** routers that have OMP peered with the SD-WAN **Controller-1** = $8 \times 2 = 16$
 - Number of SD-WAN **HUB** received **(INCLUDING the DEFAULT)** from other SD-WAN Controllers = $[2(1 \text{ Prefix} + 1 \text{ Default}) \text{ Prefix from HUB 1} \times 2 \text{ TLOC}] + [2(1 \text{ Prefix} + 1 \text{ Default}) \text{ Prefix from HUB 2} \times 2 \text{ TLOC}] = 8$
 - Number of SD-WAN Edge **spoke** routers that have OMP peered with the SD-WAN **Controller1** = 2
 - Now summing all five components gives us the total routes sent for **Controller1 (RIB-OUT)** = $12 + 0 + 0 + 0 + 16 = 28$ routes

How can we reduce the RIB-IN/RIB-OUT size?

Create an **AFFINITY Group** for SD-WAN Controllers, so that the number of routes sent per SD-WAN Controllers can be reduced.

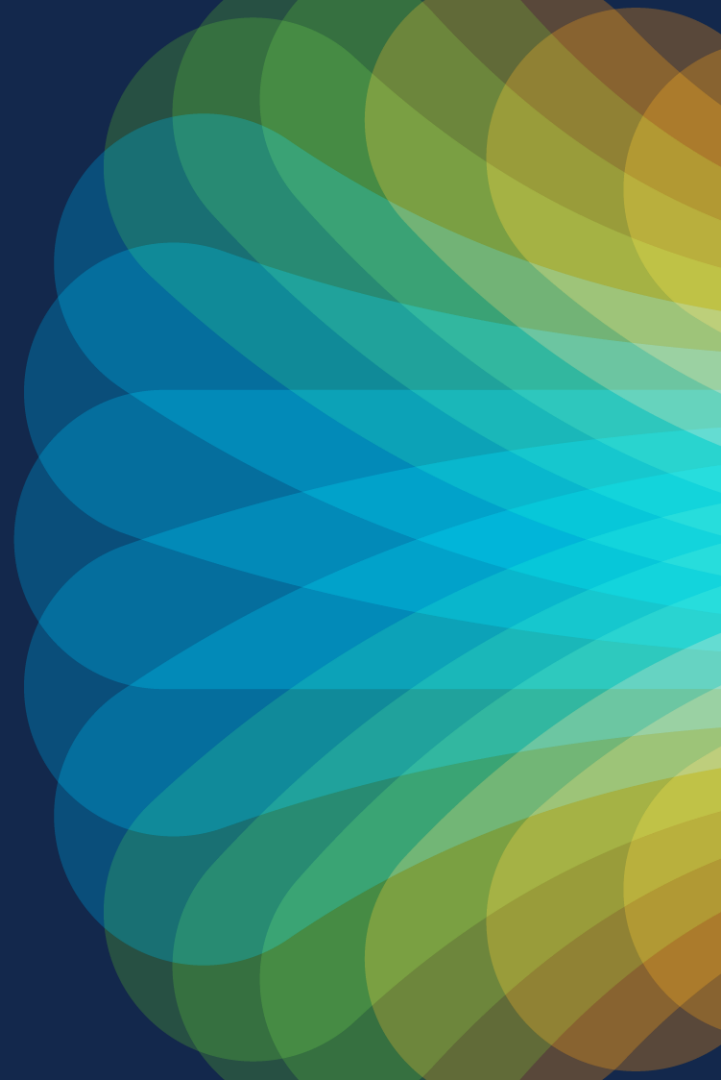
Perform Route Summarization

Add additional SD-WAN Controllers

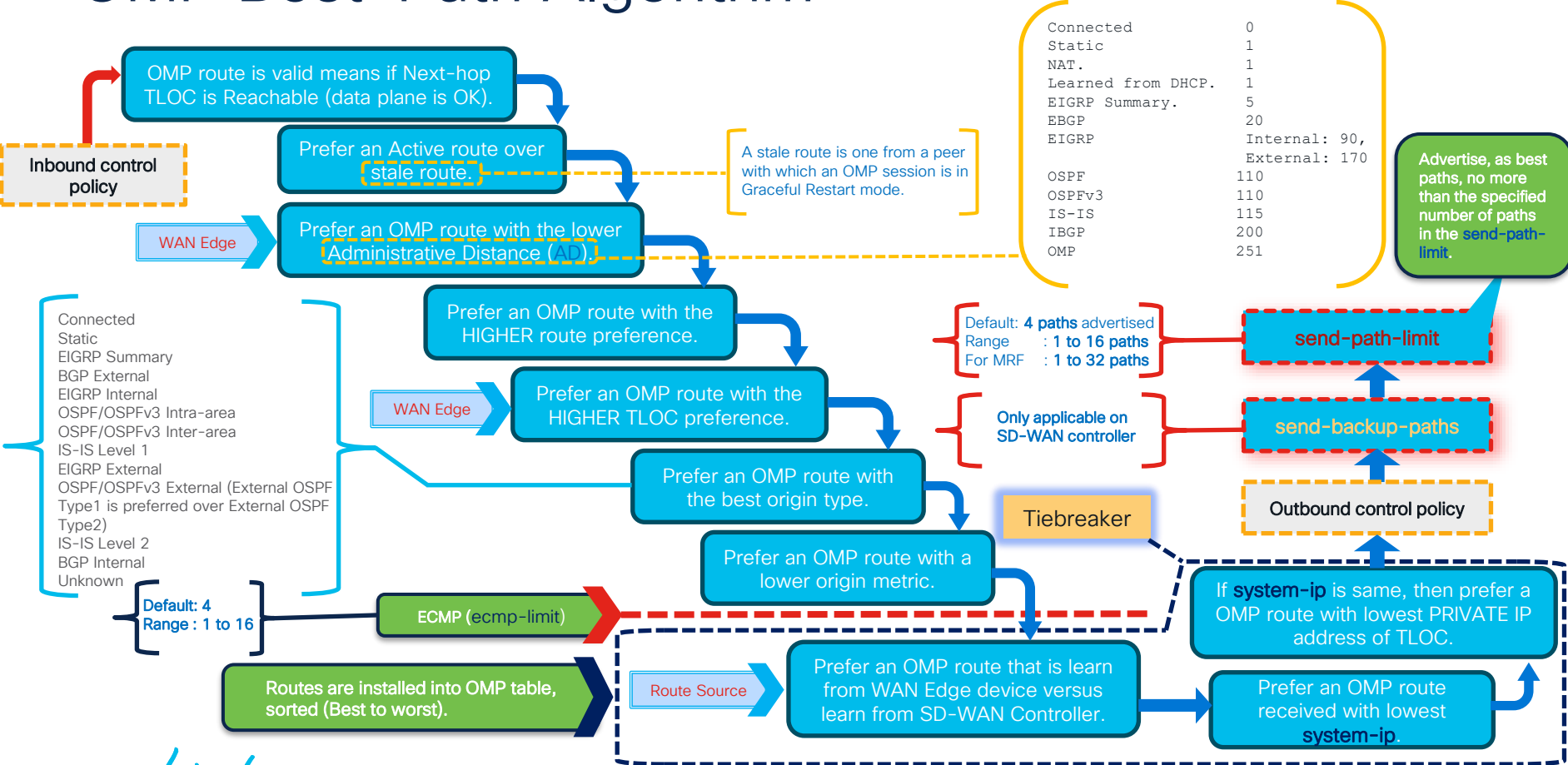


Design Considerations

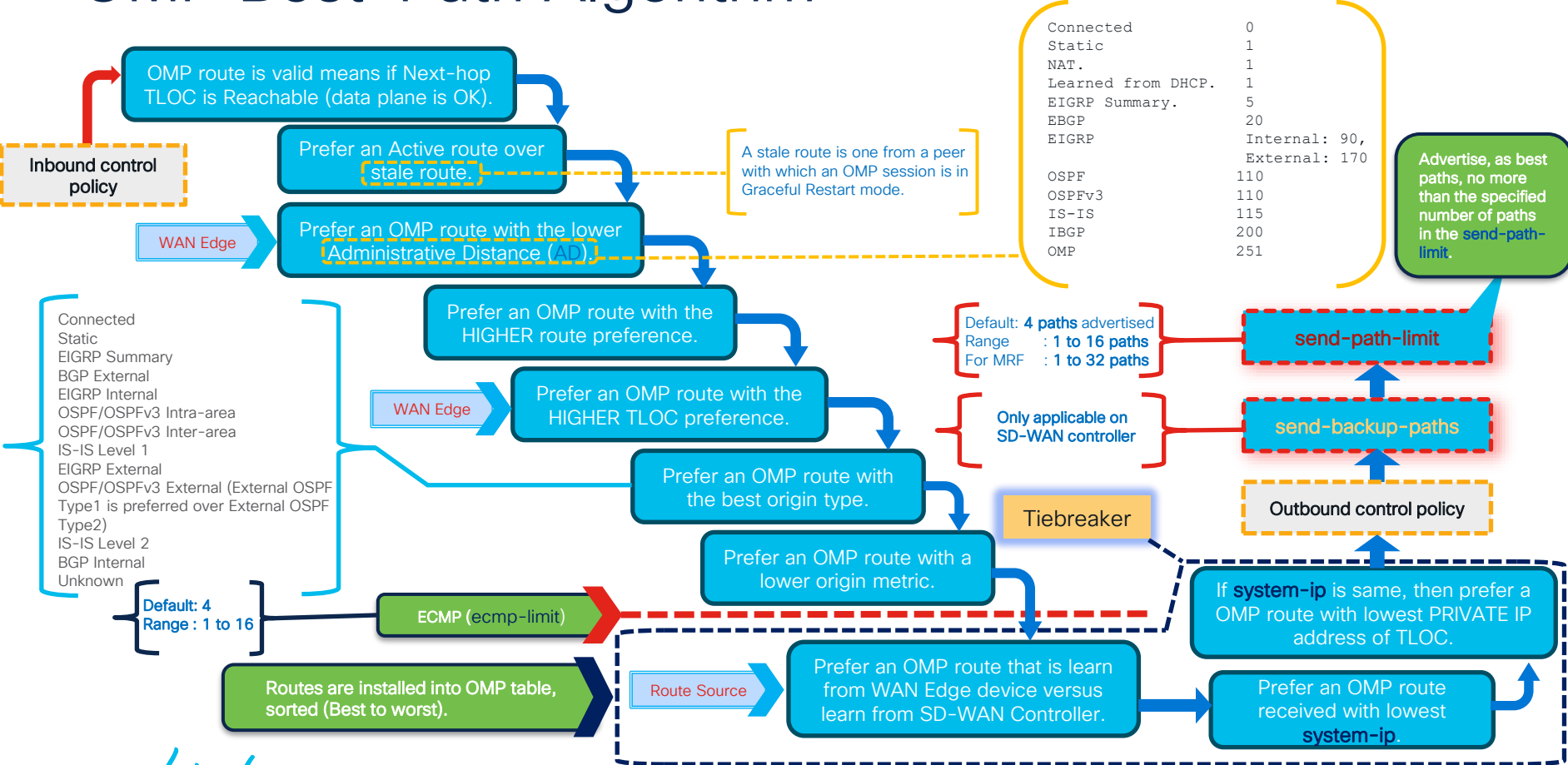
OMP Best Path selection process



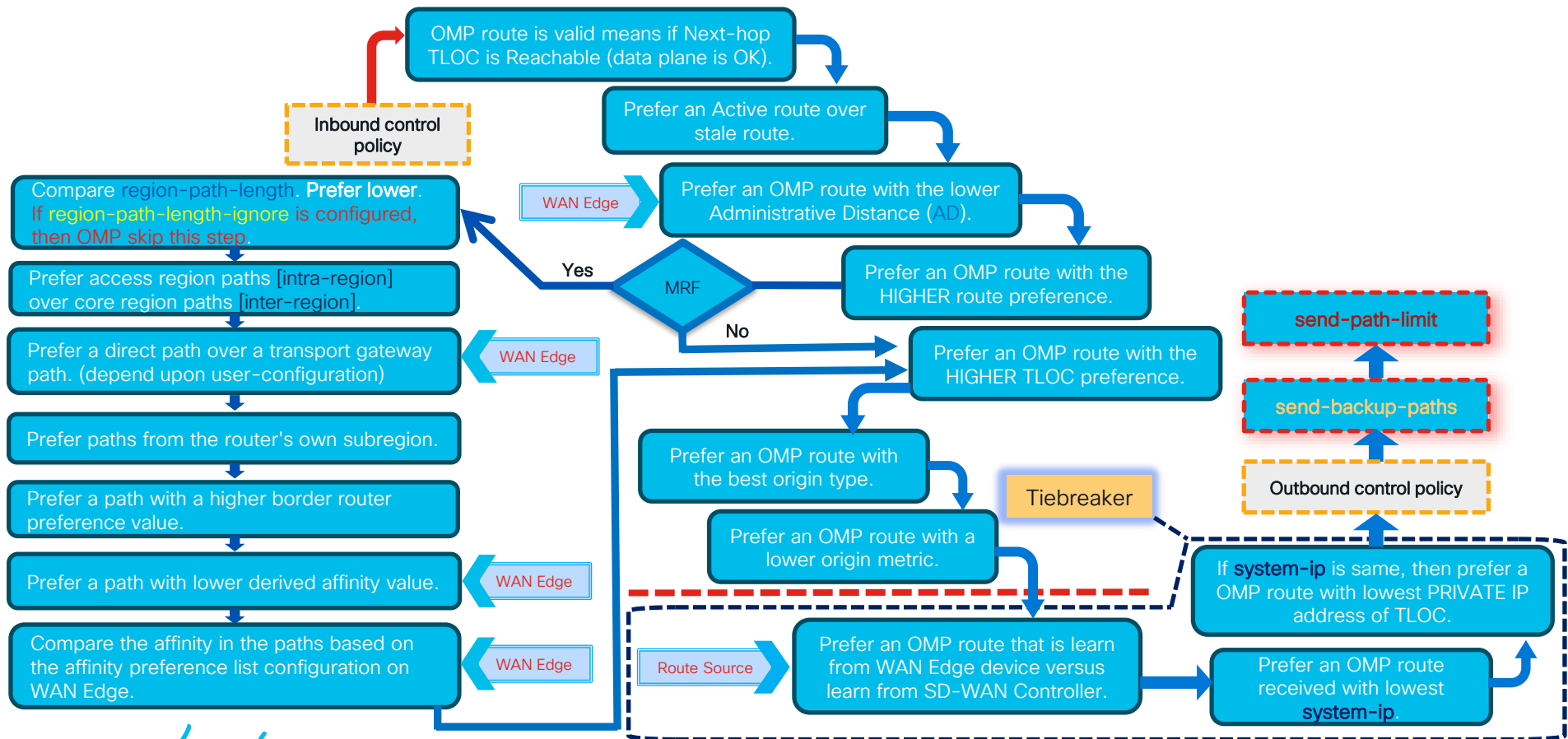
OMP Best-Path Algorithm



OMP Best-Path Algorithm



OMP Best-Path Algorithm with Multi Region Fabric(MRF)



How does OMP ensure loop avoidance?

- OMP loop avoidance is based on originator **System-IP**.
- Native built-in loop prevention mechanisms when OMP interacts with OSPF, EIGRP, RIP and BGP.

OSPF

- OSPF uses “**Down Bit**” (RFC 4577). When redistributed from OMP into OSPF on WAN Edge, it is **set**.
- When **LSA distributed** through service side network gets to the other WAN Edge, as **DN bit is set** so route is not installed into RIB on WAN Edge as **SDWAN-Dnbit** flag is set.

EIGRP

- EIGRP uses “**External Protocol**” ID field. It is set to a value of “**OMP-Agent**”.
- When the other WAN Edge on the same site receives such update, it installs the route into the EIGRP topology table, sets “**SDWAN-Down**” flag and then install into the RIB with **Administrative Distance (AD) to 252**. This, in turn, makes OMP the preferred route because it has an **AD of 251**.

RIP

- RIP uses “**OMP-ROUTE-TAG**” with value **44270**, which is **not configurable**.
- When the other WAN Edge on the same site receives RIP update, it is not get installed because of “**OMP-ROUTE-TAG**” tag. This route will get installed with **AD of 252**, **ONLY** when OMP route gets withdrawn, thus avoiding routing loop.

How does OMP ensure loop avoidance?

- OMP loop avoidance is based on originator **System-IP**.
- Native built-in loop prevention mechanisms when OMP interacts with OSPF, EIGRP, RIP and BGP.



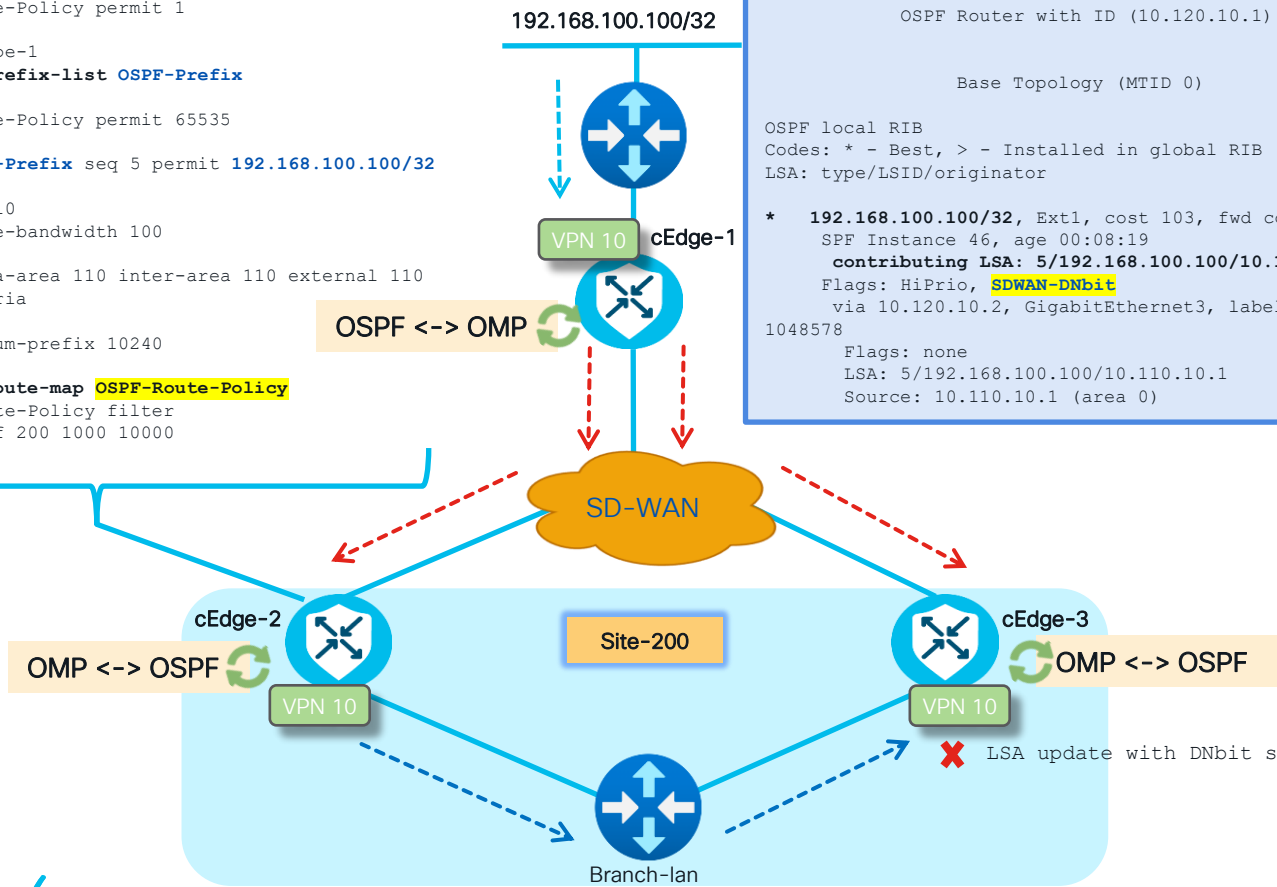
BGP

- BGP uses **SoO**, extended community which value is set to the OMP **site ID**.
- When the other WAN Edge receives the BGP update from the service-side network and there **SoO** community matches its **own site ID**, then route will **not be installed into RIB**.
- BGP peers at site must send **BGP extended communities** and have the **same site ID**.

How Down-bit (DN) works? 1/2

```

route-map OSPF-Route-Policy permit 1
  set metric 100
  set metric-type type-1
  match ip address prefix-list OSPF-Prefix
!
route-map OSPF-Route-Policy permit 65535
!
ip prefix-list OSPF-Prefix seq 5 permit 192.168.100.100/32
!
router ospf 10 vrf 10
  auto-cost reference-bandwidth 100
  compatible rfc1583
  distance ospf intra-area 110 inter-area 110 external 110
  no local-rib-criteria
  max-lsa 50000
  redistribute maximum-prefix 10240
  redistribute omp
  redistribute omp route-map OSPF-Route-Policy
  table-map OSPF-Route-Policy filter
  timers throttle spf 200 1000 10000
!
  
```



```

cEdge-3#show ip ospf rib 192.168.100.100 255.255.255.255

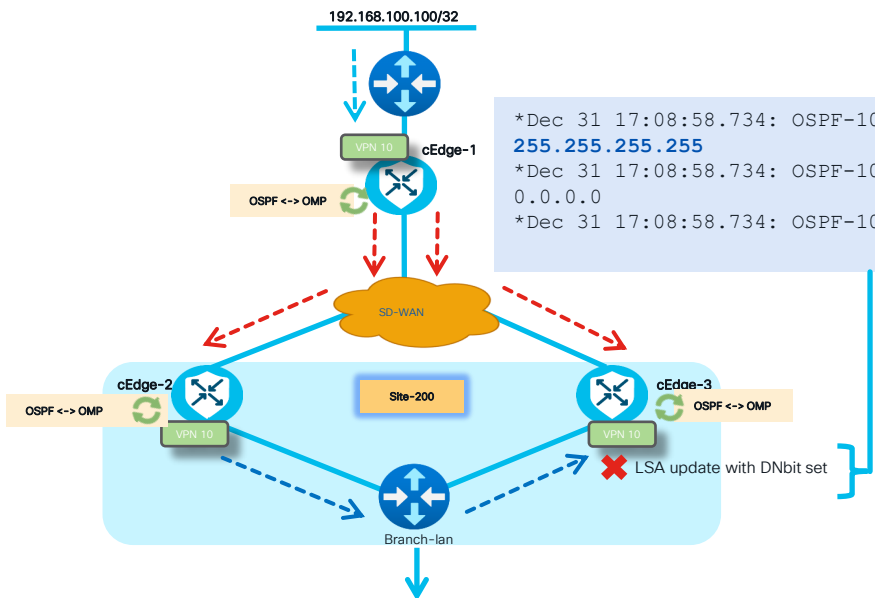
      OSPF Router with ID (10.120.10.1) (Process ID 10)

      Base Topology (MTID 0)

OSPF local RIB
Codes: * - Best, > - Installed in global RIB
LSA: type/LSID/originator

* 192.168.100.100/32, Ext1, cost 103, fwd cost 3, tag 0
  SPF Instance 46, age 00:08:19
  contributing LSA: 5/192.168.100.100/10.110.10.1
  Flags: HiPrio, SDWAN-DNbit
  via 10.120.10.2, GigabitEthernet3, label 1048578, strict label
1048578
  Flags: none
  LSA: 5/192.168.100.100/10.110.10.1
  Source: 10.110.10.1 (area 0)
  
```

How Down-bit (DN) works? 2/2



```
*Dec 31 17:08:58.734: OSPF-10 EXTER: Start processing AS External LSA 5/192.168.100.100/10.110.10.1, mask 255.255.255.255
*Dec 31 17:08:58.734: OSPF-10 EXTER: age 885, seq 0x80000028, lsa_metric 100, metric-type 1, fw-addr 0.0.0.0
*Dec 31 17:08:58.734: OSPF-10 EXTER: Downward bit with SDWAN, ignoring the LSA
```

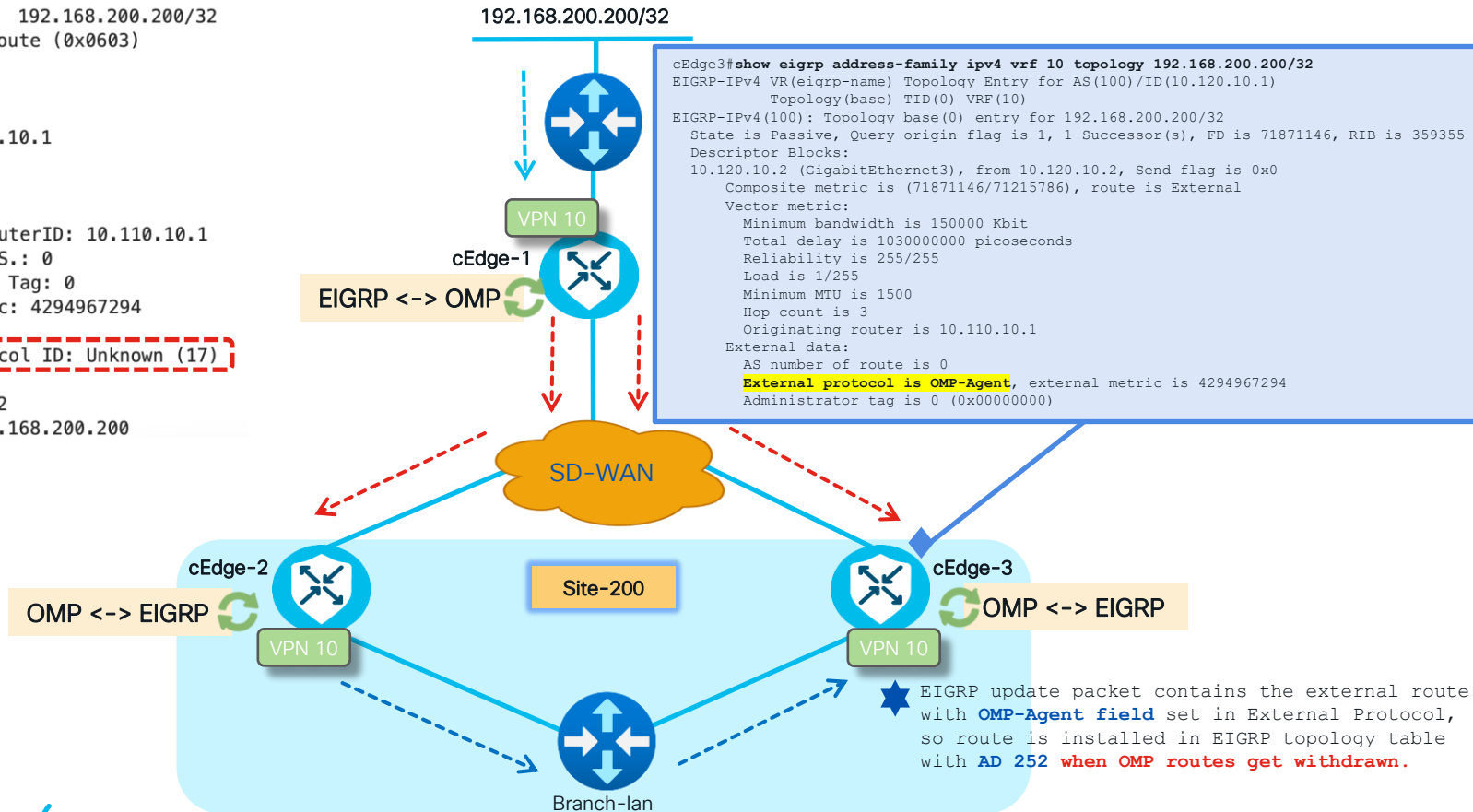
LSA-type 5 (AS-External-LSA (ASBR)), len 36

```
.000 0000 1110 0101 = LS Age (seconds): 229
0... .. = Do Not Age Flag: 0
Options: 0xa0, DN, (DC) Demand Circuits
1... .. = DN: Set
.0... .. = (U) Upaque: Not set
..1... .. = (DC) Demand Circuits: Supported
...0... .. = (L) LLS Data block: Not Present
... 0... = (N) NSSA: Not supported
... .0... = (MC) Multicast: Not capable
... ..0... = (E) External Routing: Not capable
... ..0... = (MT) Multi-Topology Routing: No
LS Type: AS-External-LSA (ASBR) (5)
Link State ID: 192.168.100.100
Advertising Router: 10.110.10.1
Sequence Number: 0x80000028
Checksum: 0xa9c9
Length: 36
Netmask: 255.255.255.255
0... .. = External Type: Type 1 (metric is specified in the same units as interface cost)
.000 0000 = TOS: 0
Metric: 100
Forwarding Address: 0.0.0.0
External Route Tag: 0
```

```
Branch-Lan#show ip route 192.168.100.100 255.255.255.255
Routing entry for 192.168.100.100/32
Known via "ospf 10", distance 110, metric 101, type extern 1
Last update from 10.110.10.1 on GigabitEthernet0/3, 03:35:49 ago
Routing Descriptor Blocks:
* 10.110.10.1, from 10.110.10.1, 03:35:49 ago, via GigabitEthernet0/3
Route metric is 101, traffic share count is 1
```

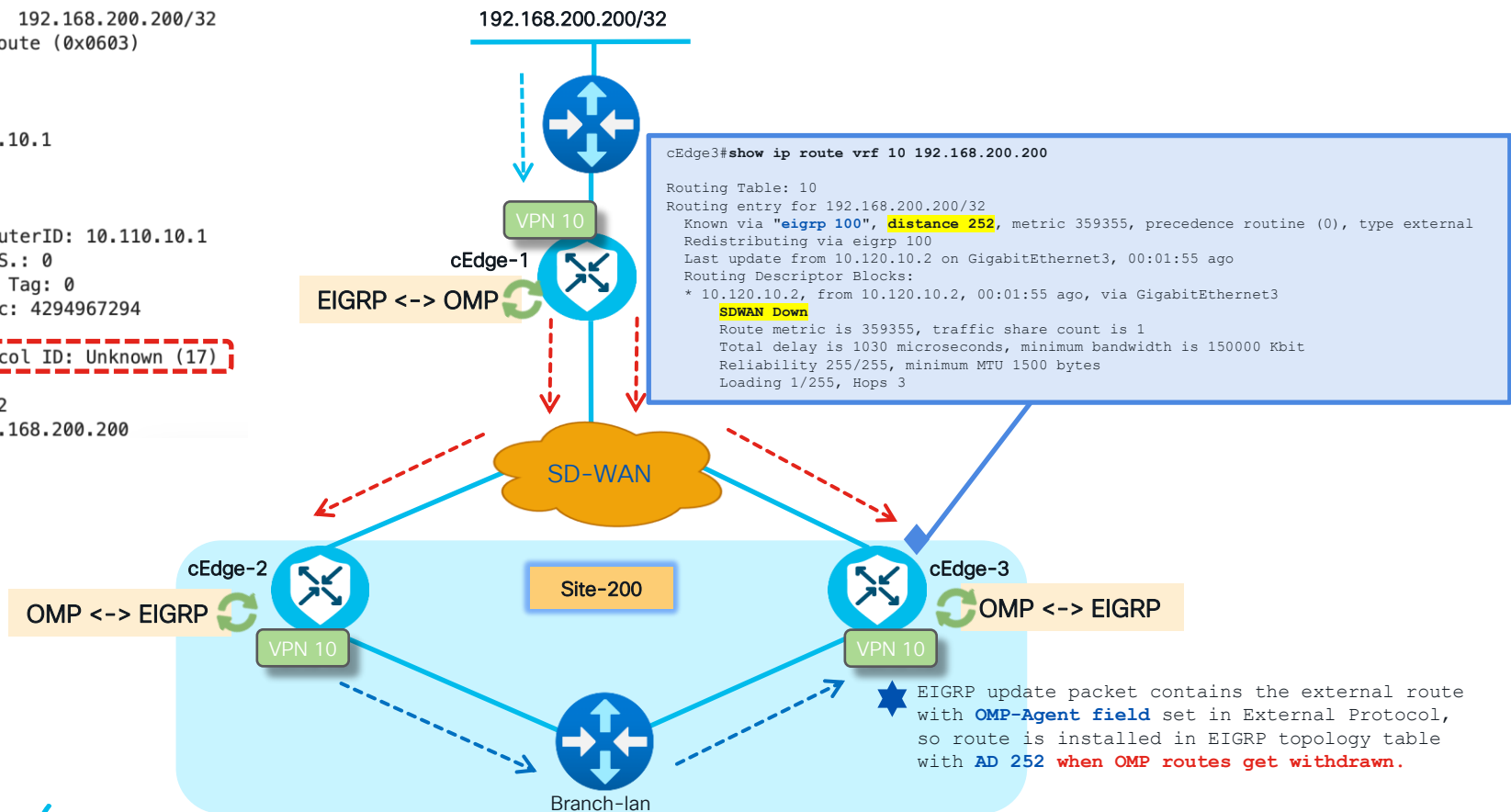
How loop avoidance works in EIGRP 1/2?

```
External Route = 192.168.200.200/32
Type: External Route (0x0603)
Length: 65
Topology: 0
AFI: IPv4 (1)
RouterID: 10.110.10.1
Wide Metric
NextHop: 0.0.0.0
External Data
  Originating RouterID: 10.110.10.1
  Originating A.S.: 0
  Administrative Tag: 0
  External Metric: 4294967294
  Reserved: 0
  External Protocol ID: Unknown (17)
External Flags
Prefix Length: 32
Destination: 192.168.200.200
```



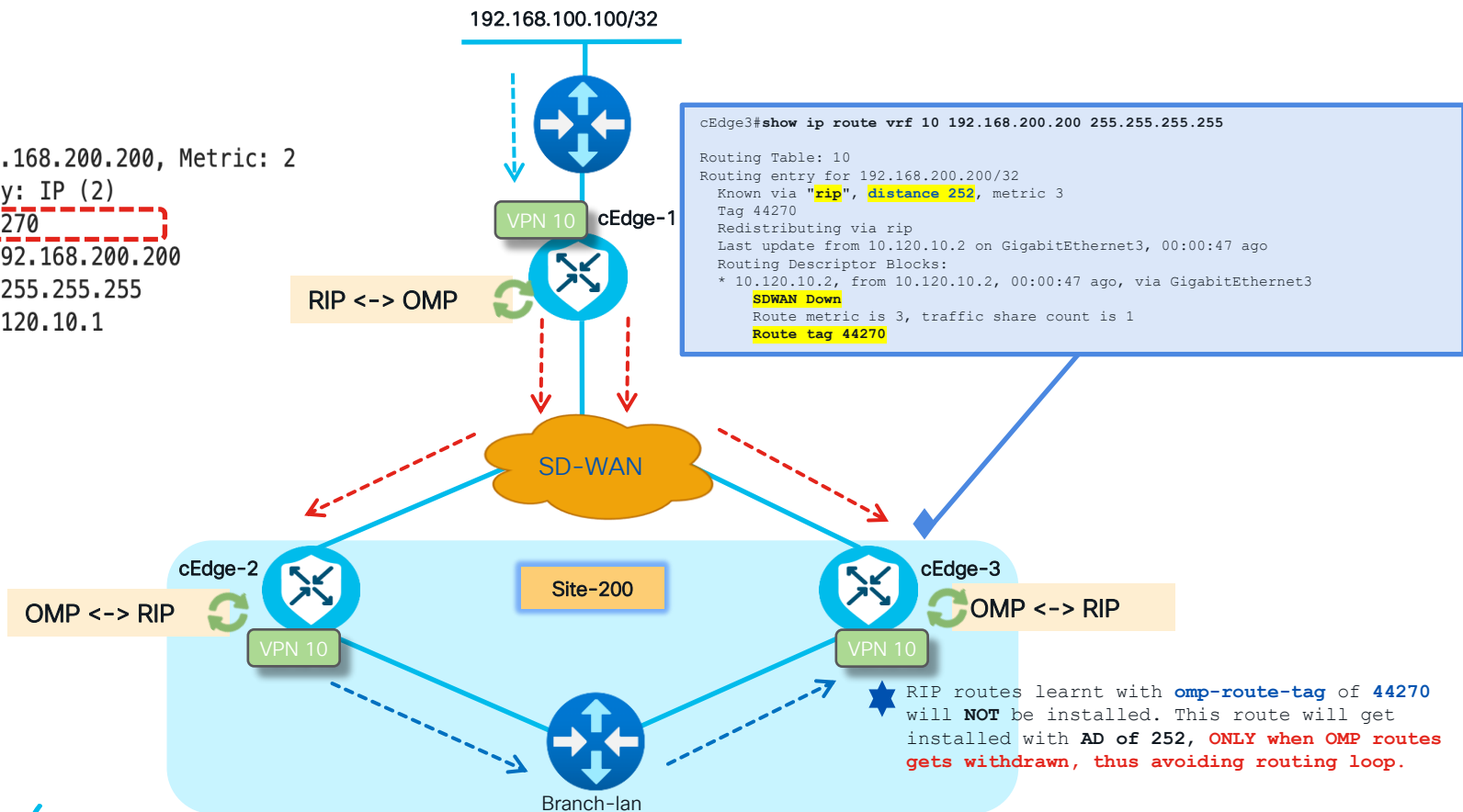
How loop avoidance works in EIGRP 2/2?

```
External Route = 192.168.200.200/32
Type: External Route (0x0603)
Length: 65
Topology: 0
AFI: IPv4 (1)
RouterID: 10.110.10.1
Wide Metric
NextHop: 0.0.0.0
External Data
  Originating RouterID: 10.110.10.1
  Originating A.S.: 0
  Administrative Tag: 0
  External Metric: 4294967294
  Reserved: 0
  External Protocol ID: Unknown (17)
External Flags
Prefix Length: 32
Destination: 192.168.200.200
```



How loop avoidance works in RIP?

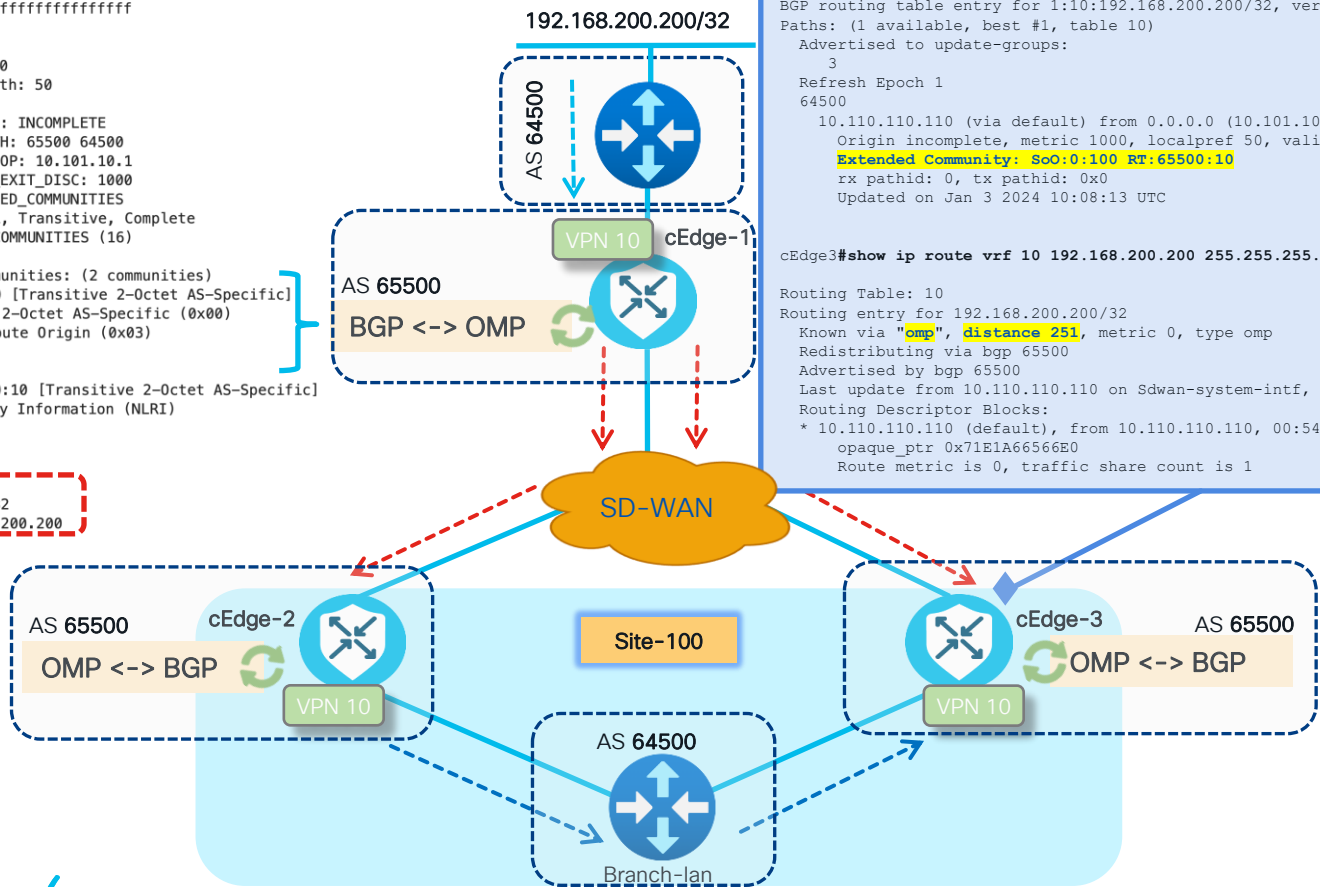
- IP Address: 192.168.200.200, Metric: 2
- Address Family: IP (2)
- Route Tag: 44270
- IP Address: 192.168.200.200
- Netmask: 255.255.255.255
- Next Hop: 10.120.10.1
- Metric: 2



How loop avoidance works in BGP 1/2?

```

Border Gateway Protocol - UPDATE Message
Marker: ffffffffffffffffffffffffffffffff
Length: 92
Type: UPDATE Message (2)
Withdrawn Routes Length: 0
Total Path Attribute Length: 50
Path attributes
  Path Attribute - ORIGIN: INCOMPLETE
  Path Attribute - AS_PATH: 65500 64500
  Path Attribute - NEXT_HOP: 10.101.10.1
  Path Attribute - MULTI_EXIT_DISC: 1000
  Path Attribute - EXTENDED_COMMUNITIES
    Flags: 0xc0, Optional, Transitive, Complete
    Type Code: EXTENDED_COMMUNITIES (16)
    Length: 16
  Carried extended communities: (2 communities)
    Route Origin: 0:100 [Transitive 2-Octet AS-Specific]
      Type: Transitive 2-Octet AS-Specific (0x00)
      Subtype (AS2): Route Origin (0x03)
      2-Octet AS: 0
      4-Octet AN: 100
    Route Target: 65500:10 [Transitive 2-Octet AS-Specific]
Network Layer Reachability Information (NLRI)
  10.0.100.0/24
  10.110.10.101/32
  10.110.10.102/32
  192.168.200.200/32
    NLRI prefix length: 32
    NLRI prefix: 192.168.200.200
  
```



```

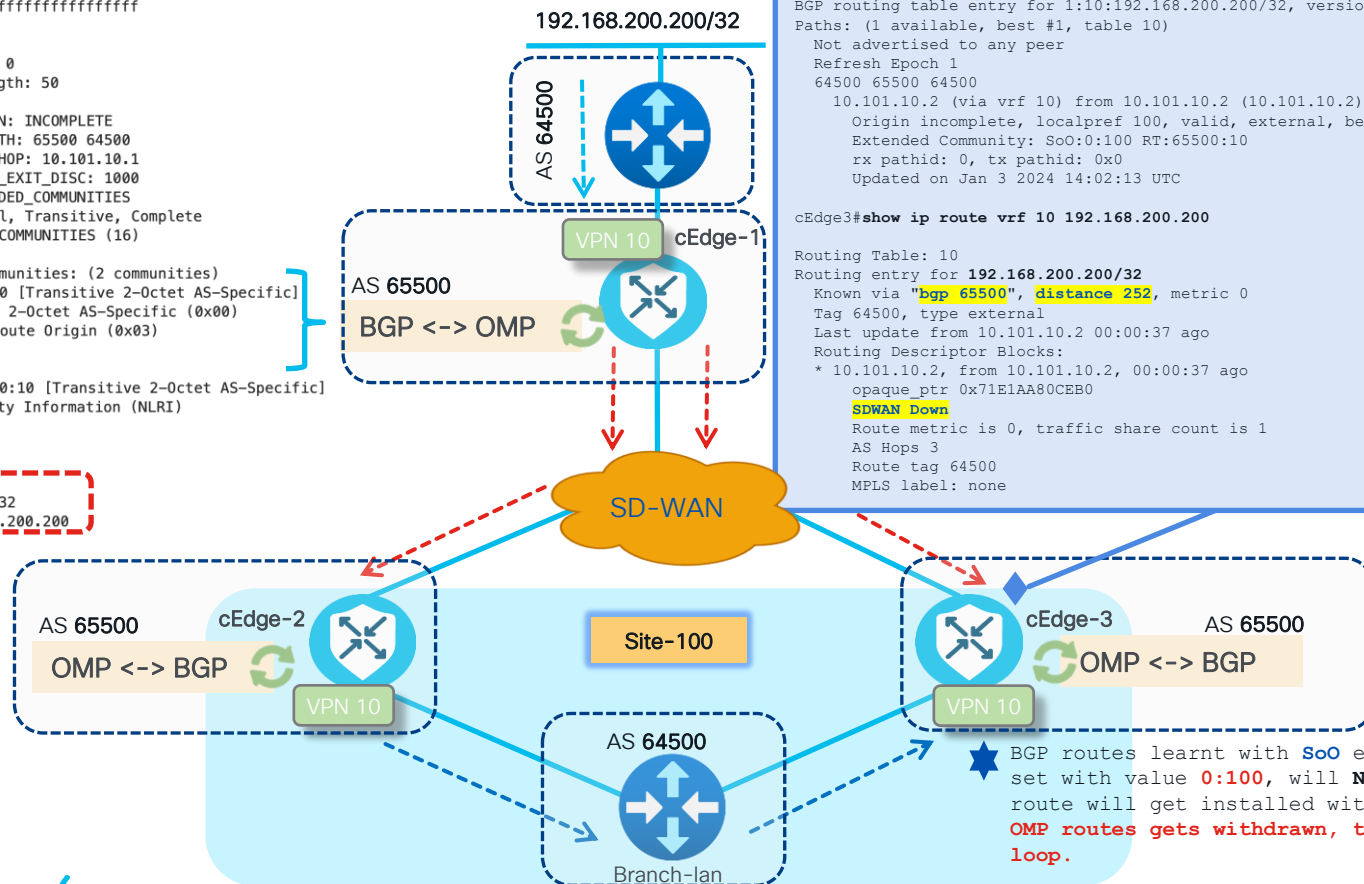
cEdge3#show bgp vpnv4 unicast vrf 10 192.168.200.200 255.255.255.255
BGP routing table entry for 1:10:192.168.200.200/32, version 265
Paths: (1 available, best #1, table 10)
Advertised to update-groups:
  3
Refresh Epoch 1
64500
  10.110.110.110 (via default) from 0.0.0.0 (10.101.10.1)
  Origin incomplete, metric 1000, localpref 50, valid, sourced, best
  Extended Community: SoO:0:100 RT:65500:10
  rx pathid: 0, tx pathid: 0x0
  Updated on Jan 3 2024 10:08:13 UTC

cEdge3#show ip route vrf 10 192.168.200.200 255.255.255.255
Routing Table: 10
Routing entry for 192.168.200.200/32
Known via "omp", distance 251, metric 0, type omp
Redistributing via bgp 65500
Advertised by bgp 65500
Last update from 10.110.110.110 on Sdwan-system-intf, 00:54:37 ago
Routing Descriptor Blocks:
* 10.110.110.110 (default), from 10.110.110.110, 00:54:37 ago, via Sdwan-system-intf
  opaque_ptr 0x71E1A66566E0
  Route metric is 0, traffic share count is 1
  
```

How loop avoidance works in BGP 2/2?

```

Border Gateway Protocol - UPDATE Message
Marker: ffffffffffffffffffffffffffffffff
Length: 92
Type: UPDATE Message (2)
Withdrawn Routes Length: 0
Total Path Attribute Length: 50
Path attributes
  > Path Attribute - ORIGIN: INCOMPLETE
  > Path Attribute - AS_PATH: 65500 64500
  > Path Attribute - NEXT_HOP: 10.101.10.1
  > Path Attribute - MULTI_EXIT_DISC: 1000
  > Path Attribute - EXTENDED_COMMUNITIES
    > Flags: 0xc0, Optional, Transitive, Complete
    Type Code: EXTENDED_COMMUNITIES (16)
    Length: 16
    Carried extended communities: (2 communities)
  > Route Origin: 0:100 [Transitive 2-Octet AS-Specific]
    > Type: Transitive 2-Octet AS-Specific (0x00)
    Subtype (AS2): Route Origin (0x03)
    2-Octet AS: 0
    4-Octet AN: 100
  > Route Target: 65500:10 [Transitive 2-Octet AS-Specific]
Network Layer Reachability Information (NLRI)
  > 10.0.100.0/24
  > 10.110.10.101/32
  > 10.110.10.102/32
  > 192.168.200.200/32
    NLRI prefix length: 32
    NLRI prefix: 192.168.200.200
  
```



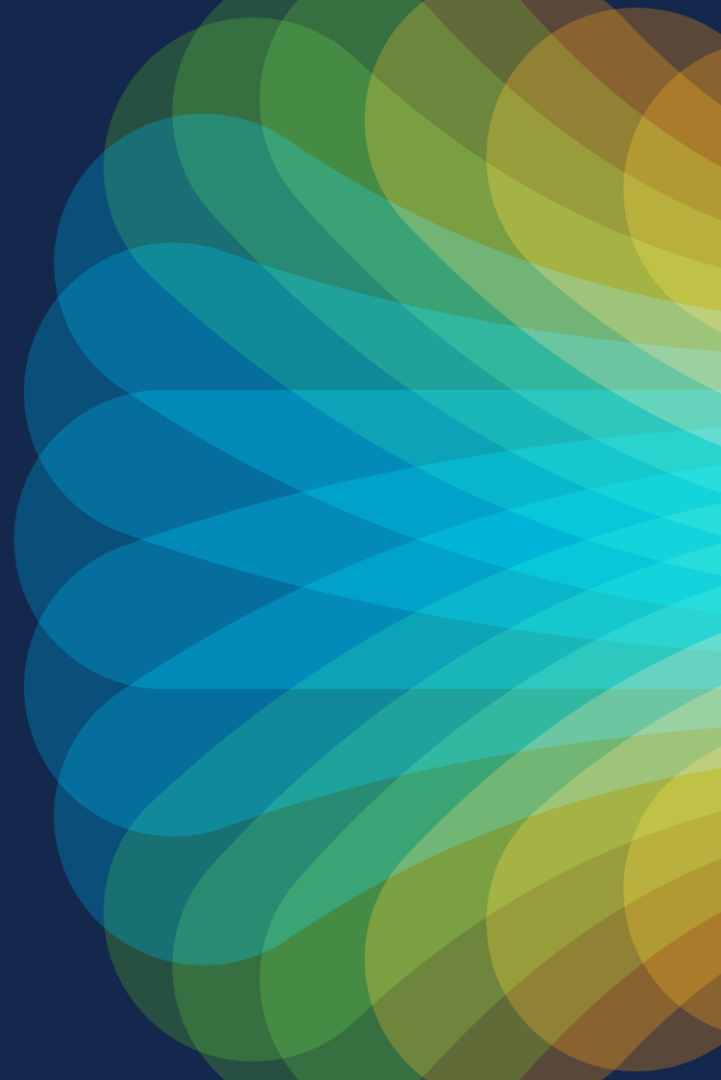
```

cEdge3#show bgp vpnv4 unicast vrf 10 192.168.200.200
BGP routing table entry for 1:10:192.168.200.200/32, version 379
Paths: (1 available, best #1, table 10)
Not advertised to any peer
Refresh Epoch 1
64500 65500 64500
  10.101.10.2 (via vrf 10) from 10.101.10.2 (10.101.10.2)
  Origin incomplete, localpref 100, valid, external, best
  Extended Community: SoO:0:100 RT:65500:10
  rx pathid: 0, tx pathid: 0x0
  Updated on Jan 3 2024 14:02:13 UTC

cEdge3#show ip route vrf 10 192.168.200.200
Routing Table: 10
Routing entry for 192.168.200.200/32
Known via "bgp 65500", distance 252, metric 0
Tag 64500, type external
Last update from 10.101.10.2 00:00:37 ago
Routing Descriptor Blocks:
* 10.101.10.2, from 10.101.10.2, 00:00:37 ago
  opaque_ptr 0x71E1AA80CEB0
  SDWAN Down
  Route metric is 0, traffic share count is 1
  AS Hops 3
  Route tag 64500
  MPLS label: none
  
```

★ BGP routes learnt with SoO ext-community attribute set with value 0:100, will NOT be installed. This route will get installed with AD of 252, ONLY when OMP routes gets withdrawn, thus avoiding routing loop.

SD-WAN Multicast



Multicast Terminologies

RP - Rendezvous Point (Root of the tree)

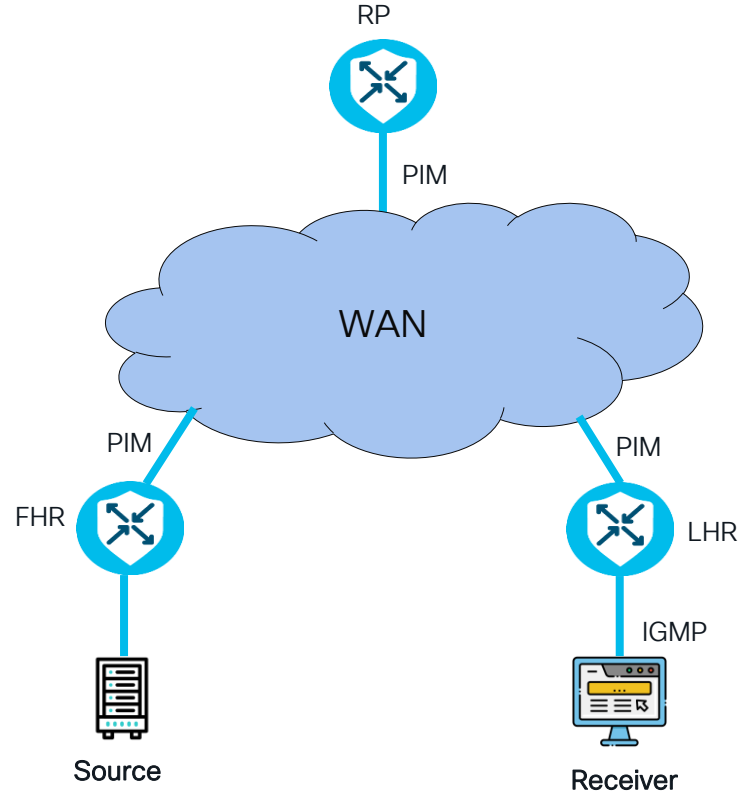
FHR - First Hop Router (Source)

LHR - Last Hop Router (Receiver)

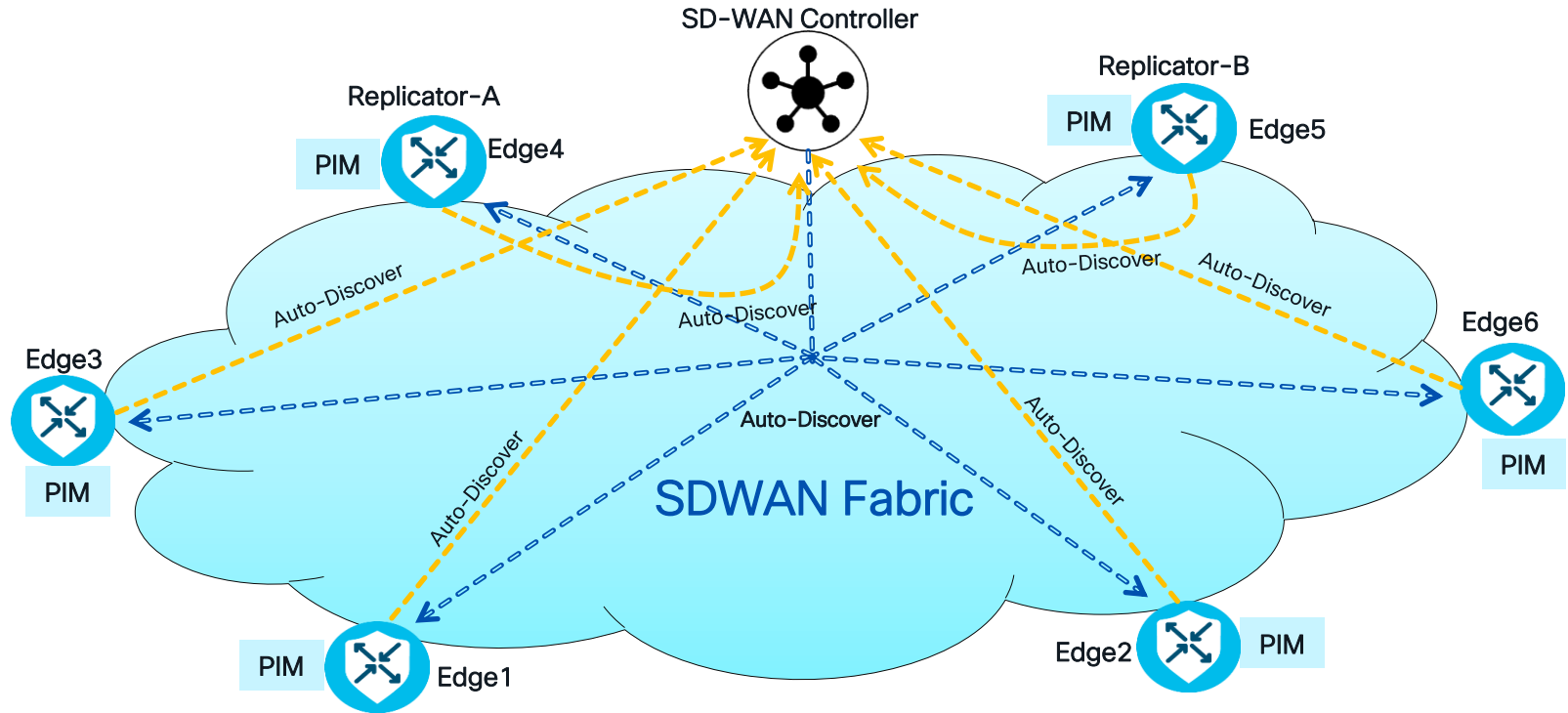
IGMP - Internet Group Management Protocol

PIM - Protocol Independent Multicast

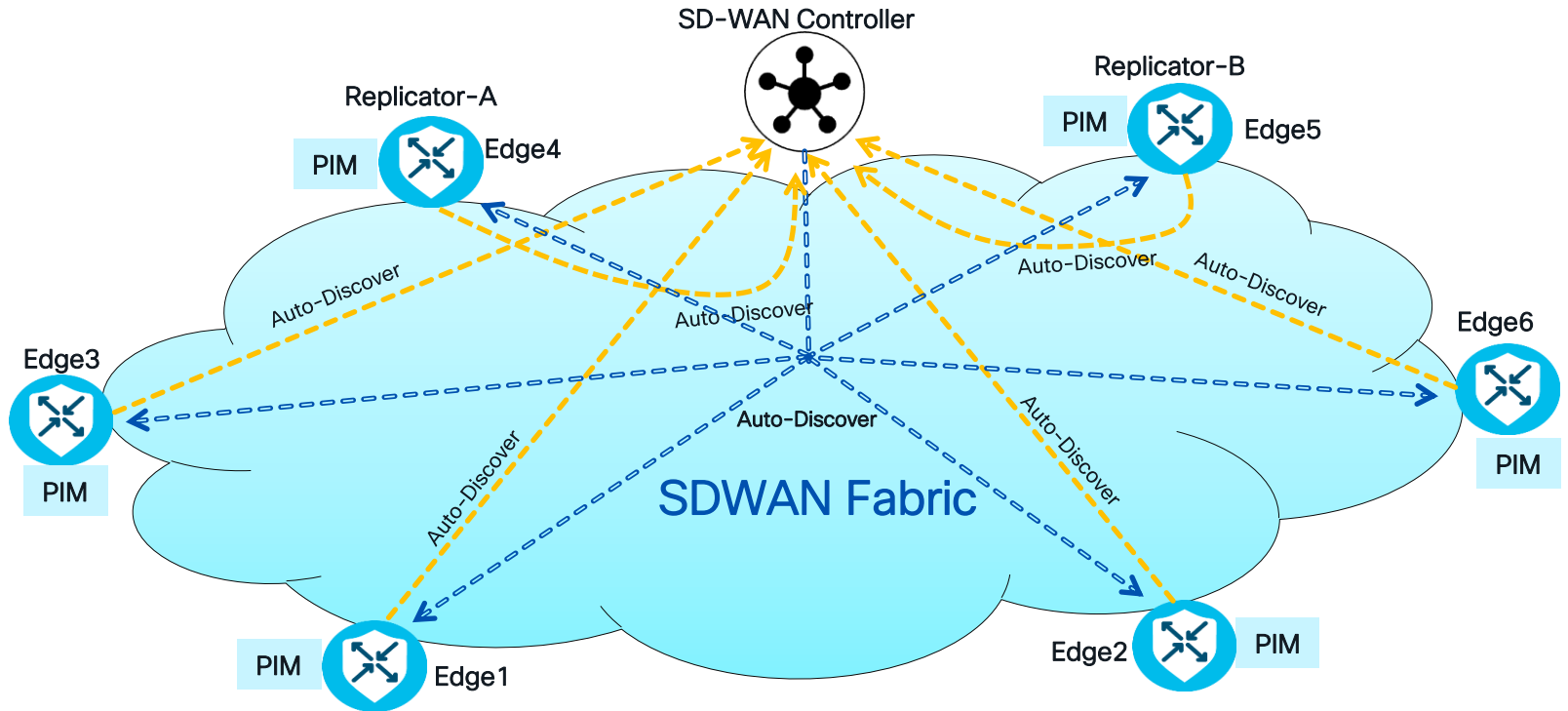
ASM - Any Source Multicast



OMP Multicast Auto-Discovery

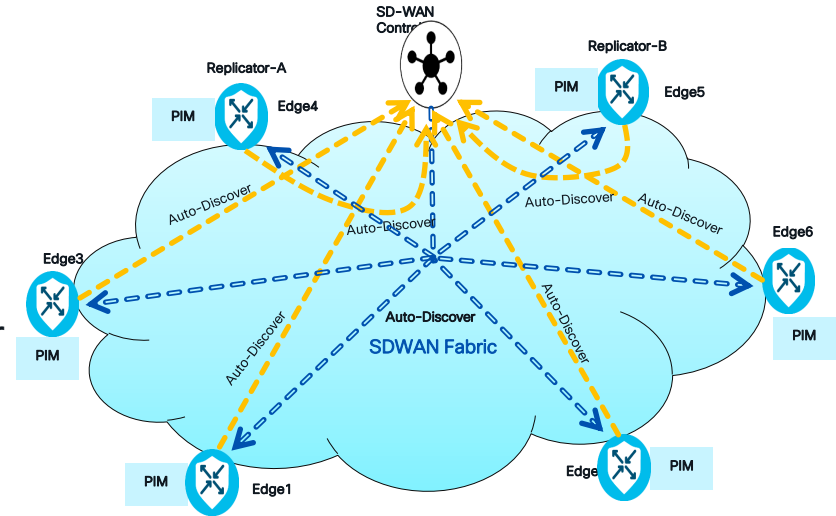


OMP Multicast Auto-Discovery



OMP Multicast Auto-Discovery

- OMP Multicast Auto-discovery packet contains:
 - List of WAN Edge devices are configured as multicast capable
 - List of devices are configured as replicator
 - Replicator capacity, GPS location etc.
- The multicast Autodiscover routes indicate whether the router has PIM enabled and whether it is a replicator.



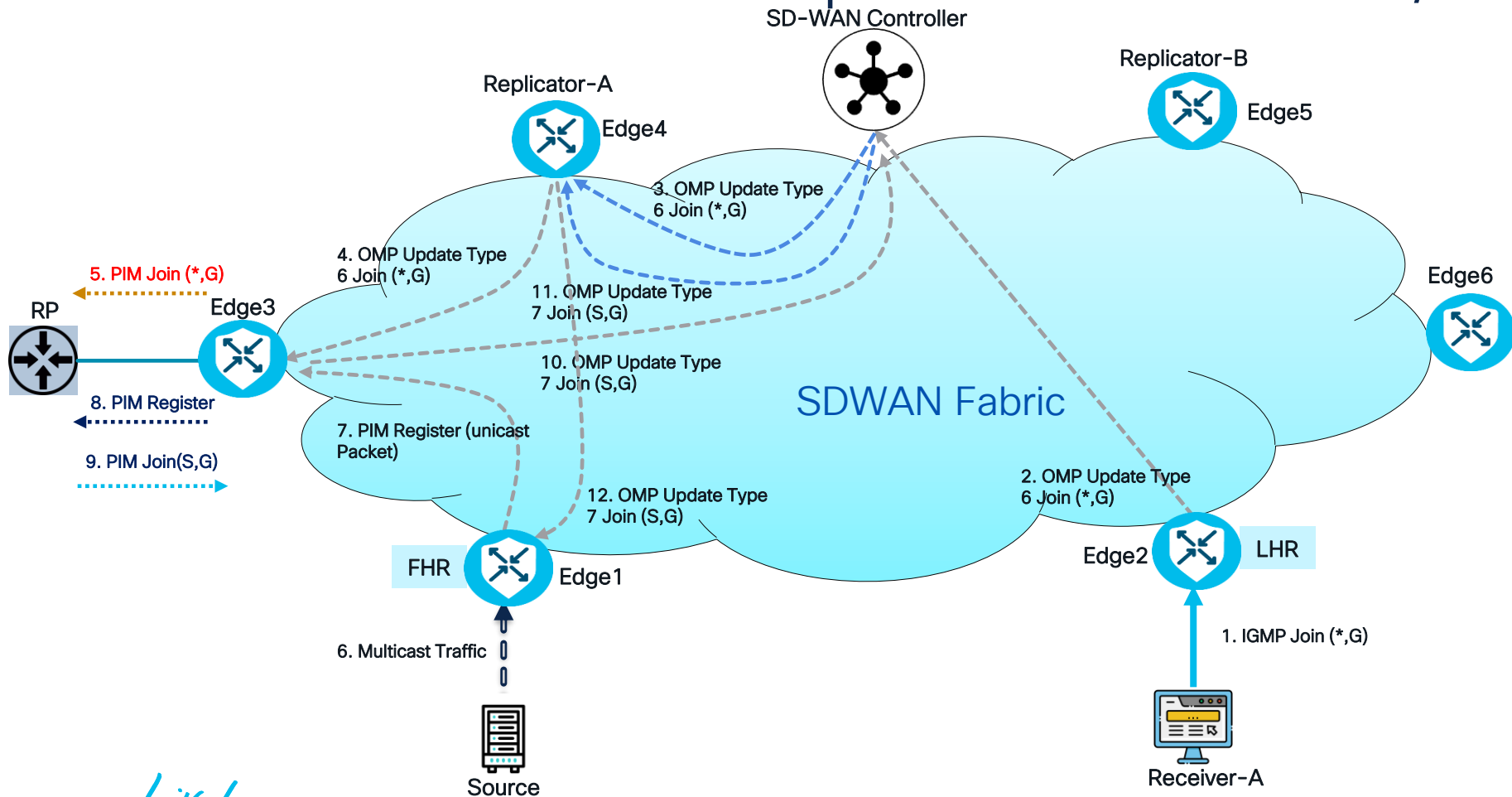
SD-WAN Multicast control packet flow - ASM

- Source register itself to an RP
- Receiver sends the (*,G) join
- First Join gets forwarded to the SD-WAN controller as an OMP packet and then forwarded to the replicator
- Replicator forwards (*,G) to the RP
- RP forwards it to the source
- Stream is forwarded to the receiver through the replicator. **Stream NEVER goes to SD-WAN controller.**
- Once receiver has the source information, it will then join using (S,G)
- First (S,G) join gets forwarded as an OMP control packet to the SD-WAN controller and then to replicator
- Replicator then forward the (S,G) to the source.

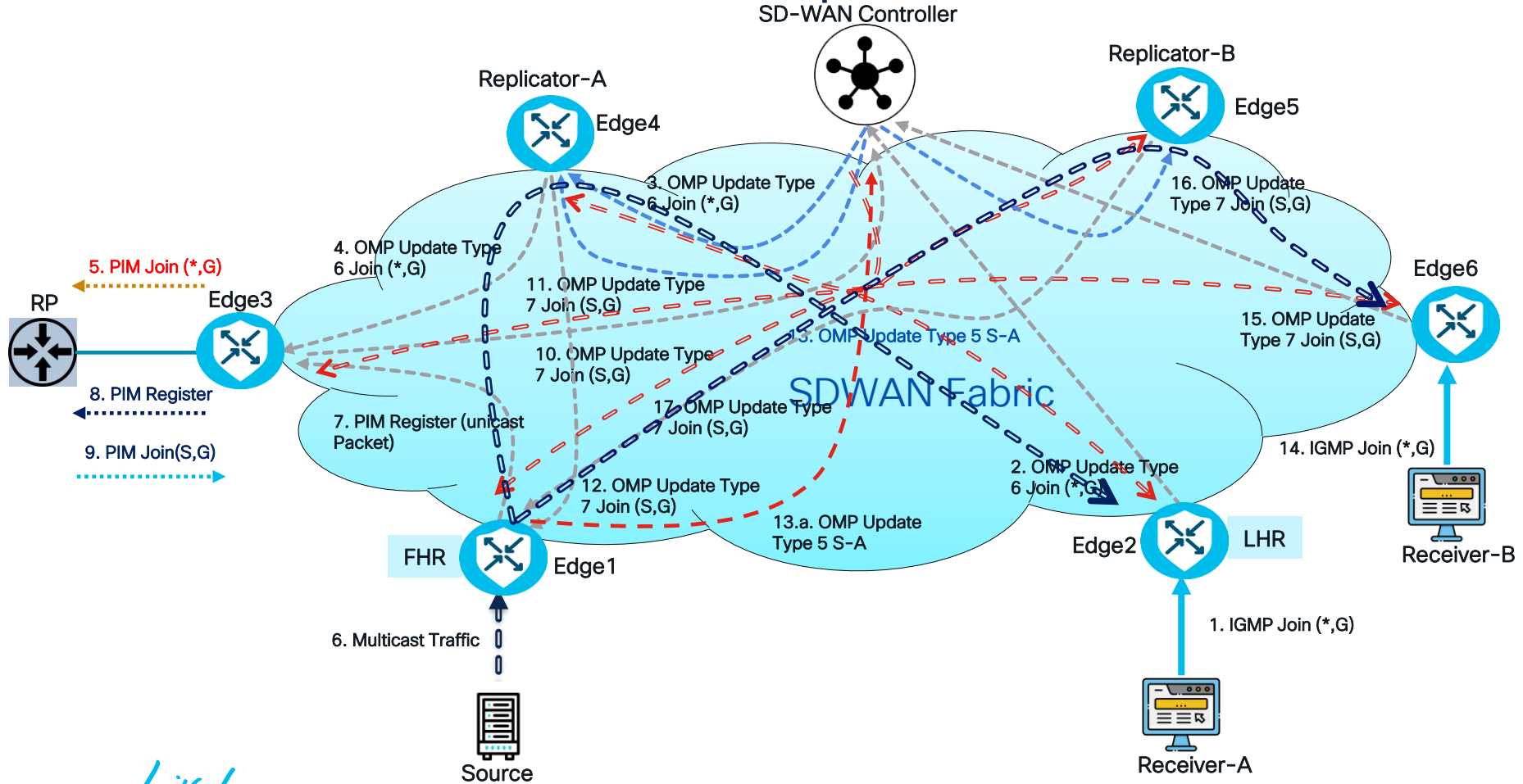
OMP Multicast Message

Route Type	OMP Overlay Multicast Route Type
5	Source Active
6	Shared Tree Join
7	Source Tree Join

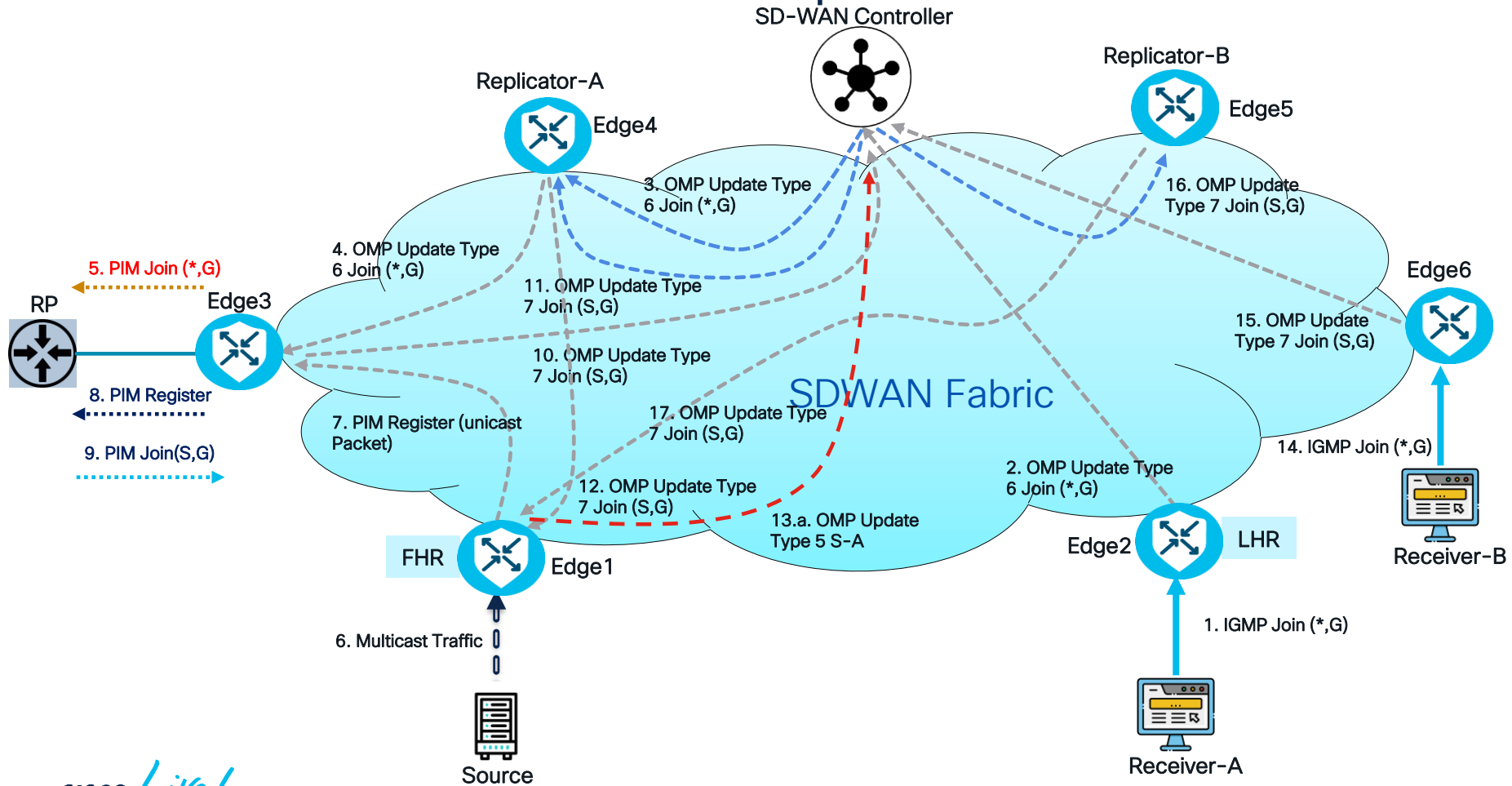
SD-WAN Multicast control packet flow - ASM 1/3



SD-WAN Multicast control packet flow - ASM 2/3



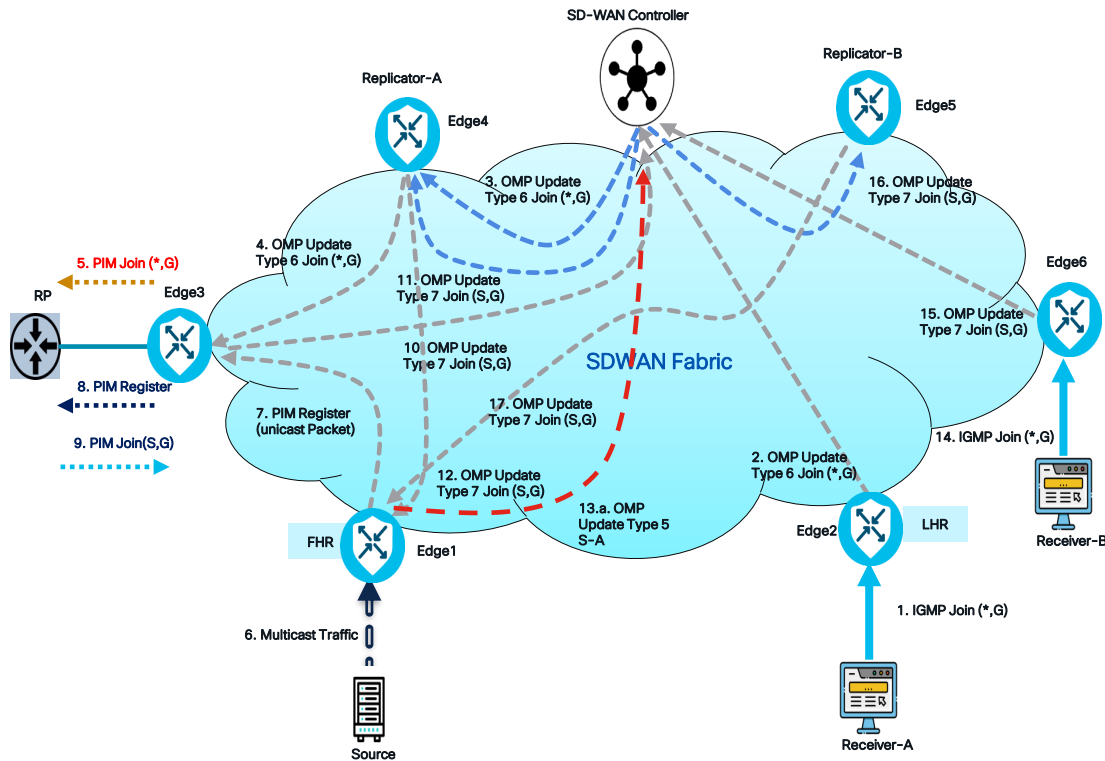
SD-WAN Multicast control packet flow - ASM 2/3



SD-WAN Multicast control packet flow - ASM 3/3



For Reference



- IGMP join from Receiver is sent to LHR
- Type 6 (*,G) OMP join from LHR is sent to SD-WAN controller and SD-WAN controller forwards it to Replicator.
- Type 6 (*,G) OMP join from Replicator is sent to RP.
- Source streams the traffic to FHR.
- FHR sends PIM Registration message to RP.
- Type 7 (S,G) join from **RP** is sent to SD-WAN controller and then SD-WAN controller send it to Replicator.
- Type 7 (S,G) join from Replicator to is sent to FHR.
- Type 5 (S,A) message is sent from FHR to SD-WAN controller and SD-WAN controller sends to all the WAN Edges including LHR, RP and Replicator.
- Type 7 (S,G) join from LHR is sent to Replicator
- FHR transmits the traffic to the Replicator, which subsequently forwards it to LHR, and ultimately, the receiver obtains the Multicast stream.

SD-WAN Multicast Key points 1/2

- The significance of replicator selection lies solely from the perspective of the **LHR** (receivers' point of view).
- When multiple replicators are involved, there is **no synchronous** mechanism in place. In the event that a replicator becomes inactive for a specific stream, the receiver will be required to re-join the tree using an alternative replicator.
- In scenarios where there are **no receivers** for the multicast traffic, a PIM Register unicast message will be sent to RP. However, since there are **no receivers at that point**, the RP will respond by transmitting a **PIM Register STOP message back to the source**.
- Enabling the “**spt-mode**” helps minimize the exchange of control plane packets for SDWAN multicast, while still allowing multicast data traffic to be replicated through the chosen replicator.
- It is crucial to **ensure that “spt-mode” is enabled on every WAN edge router** within the SDWAN fabric.

SD-WAN Multicast Key points 2/2

- Data policy interworking with SDWAN overlay multicast is **NOT officially tested** before and **not supported**.

Limitations of Multicast Configuration

Multicast overlay routing does not support the following features:

- MSDP/Anycast-RP on Cisco SD-WAN routers
- IPv6 overlay and IPv6 underlay
- Dynamic BFD tunnel for multicast
- Multicast with asymmetric unicast routing
- Multicast overlay working does not support Data Policy. In case data policy is configured, then only required traffic is matched and not multicast traffic.
- We are planning to fix in **20.15/17.15** release, with enhancement request [CSCwf85691](#), with multicast support for Hub and spoke topology.

```
data-prefix-list Unicast_IPv4
  ip-prefix 0.0.0.0/1
  ip-prefix 128.0.0.0/2
  ip-prefix 192.0.0.0/3
!
data-policy_VPN_1_AS-Set-local-tloc-policy_v2
  vpn-list VPN 1
```

```
!
action accept
  count all_traffic_counter_1526348700
  set
    local-tloc-list
    color mpls
    encap ipsec
!
!
!
default-action accept
```

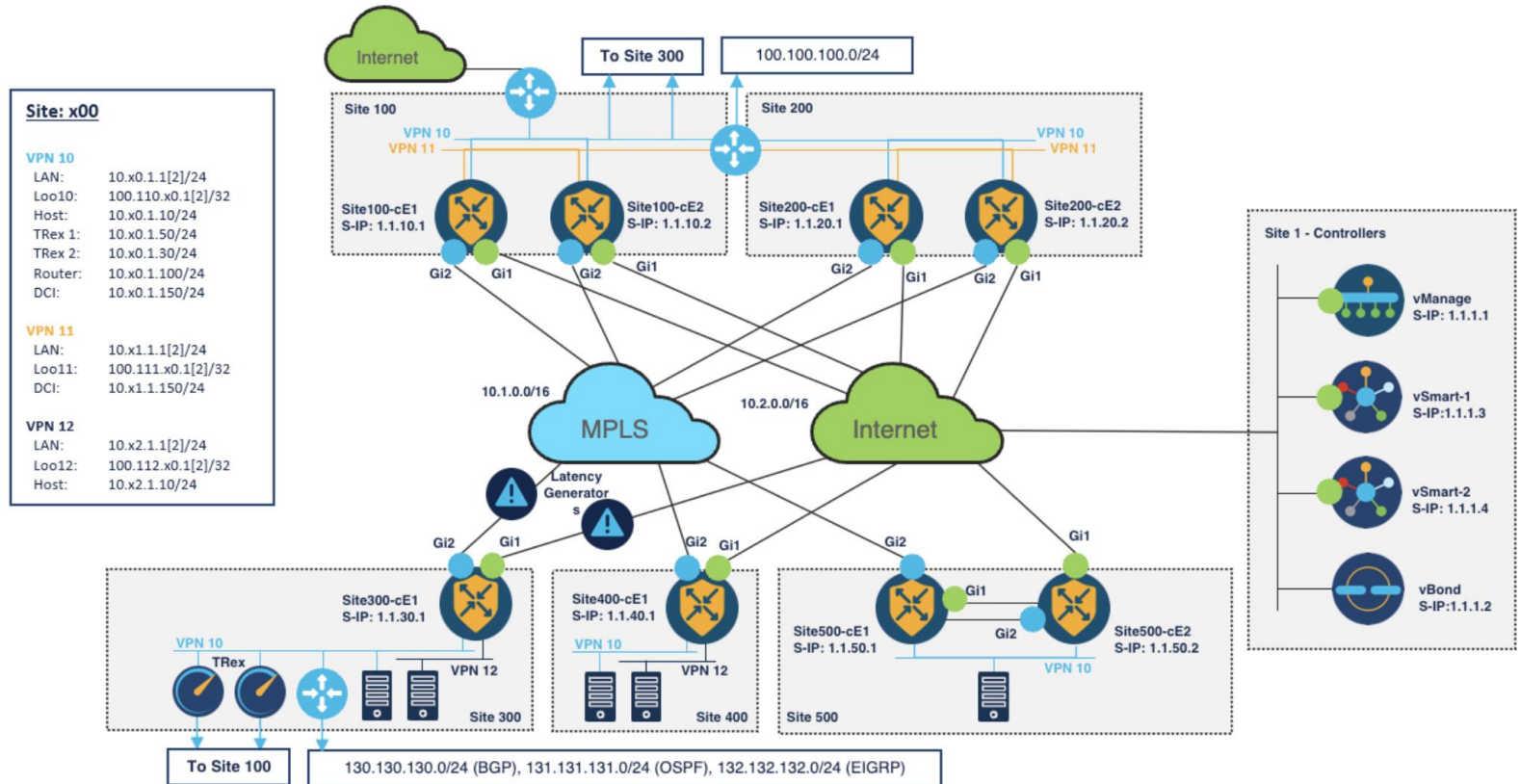
SD-WAN Multicast Key points 2/2

- Data policy interworking with SDWAN overlay multicast is **NOT officially tested** before and **not supported**.
- Even **localized QoS policy**, it might count on data policy to set different forwarding class.
- While configuring data policy or localized QoS policy, we need to make sure **ONLY unicast traffic** is matched.
- In hub and spoke topologies **where multicast senders and receivers are at spokes, multicast traffic will be dropped**.
- We are planning to fix in **20.15/17.15** release, with enhancement request **CSCwf85691**, with multicast support for Hub and spoke topology.

```
data-prefix-list Unicast_IPv4
  ip-prefix 0.0.0.0/1
  ip-prefix 128.0.0.0/2
  ip-prefix 192.0.0.0/3
!
data-policy_VPN_1_AS-Set-local-tloc-policy_v2
vpn-list VPN_1
sequence 1
match
  source-ip 0.0.0.0/0
  dscp 10
  destination-data-prefix-list Unicast_IPv4 ← MATCH ONLY Unicast Traffic
!
action accept
count dscp10_counter_1526348700
set
  local-tloc-list
  color public-internet
  encaps ipsec
!
!
!
sequence 11
match
  source-ip 0.0.0.0/0
  destination-data-prefix-list Unicast_IPv4 ← MATCH ONLY Unicast Traffic
!
action accept
count all_traffic_counter_1526348700
set
  local-tloc-list
  color mpls
  encaps ipsec
!
!
!
default-action accept
```

SD-WAN Use Case Topology

Demo Topology



Full Mesh Topology

- Overview
- Devices
- Tunnels
- Applications
- Security
- VPN
- Logs
- Multicloud

Overview

Grid, List, Location icons

CONTROLLERS

1 vBond
2 vSmart
1 vManage

WAN Edges

8 Reachable

CERTIFICATE STATUS

0 Warning

LICENSING

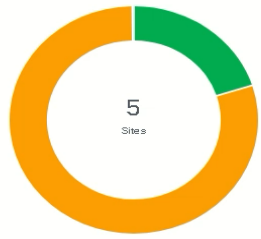
0 Assigned
8 Unassigned

REBOOT

0 Last 24 hrs

Site Health

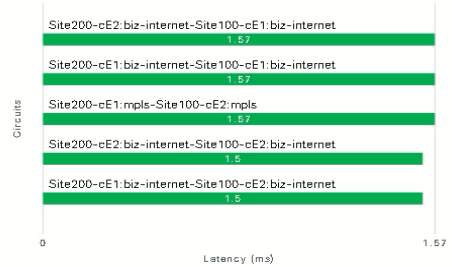
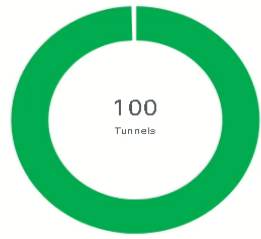
24 Hours Good Performing Sites



View Details

Tunnel Health

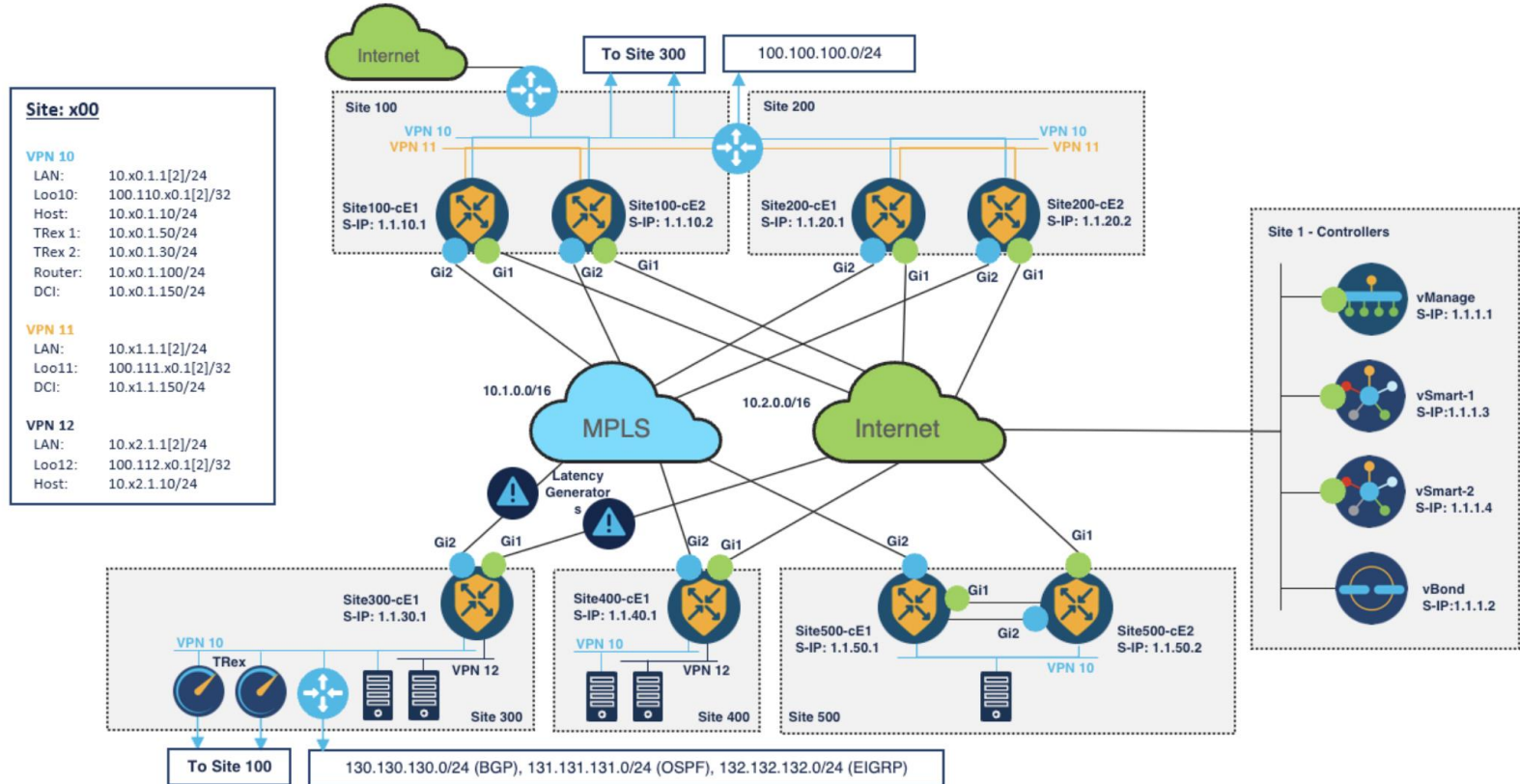
24 Hours Latency (ms) Good Tunnels



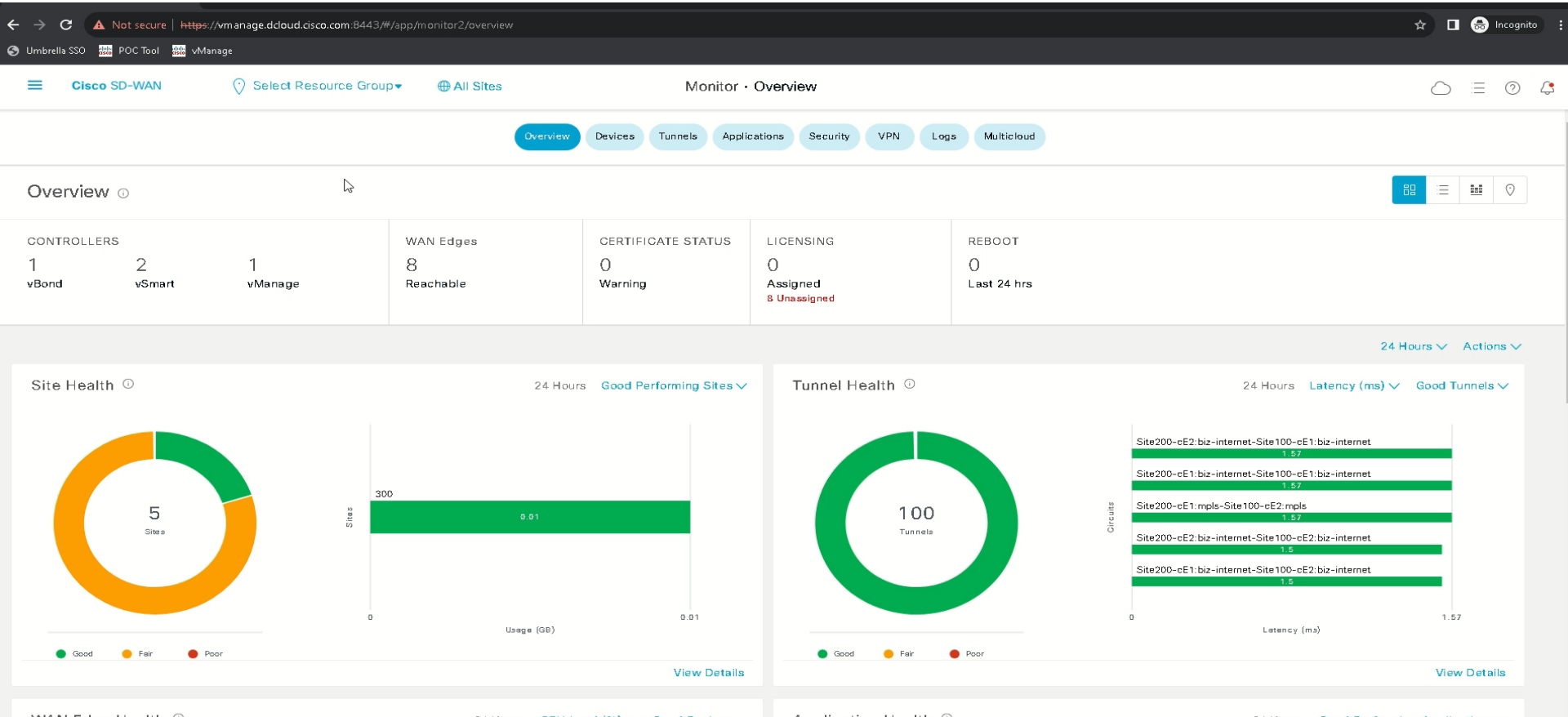
View Details



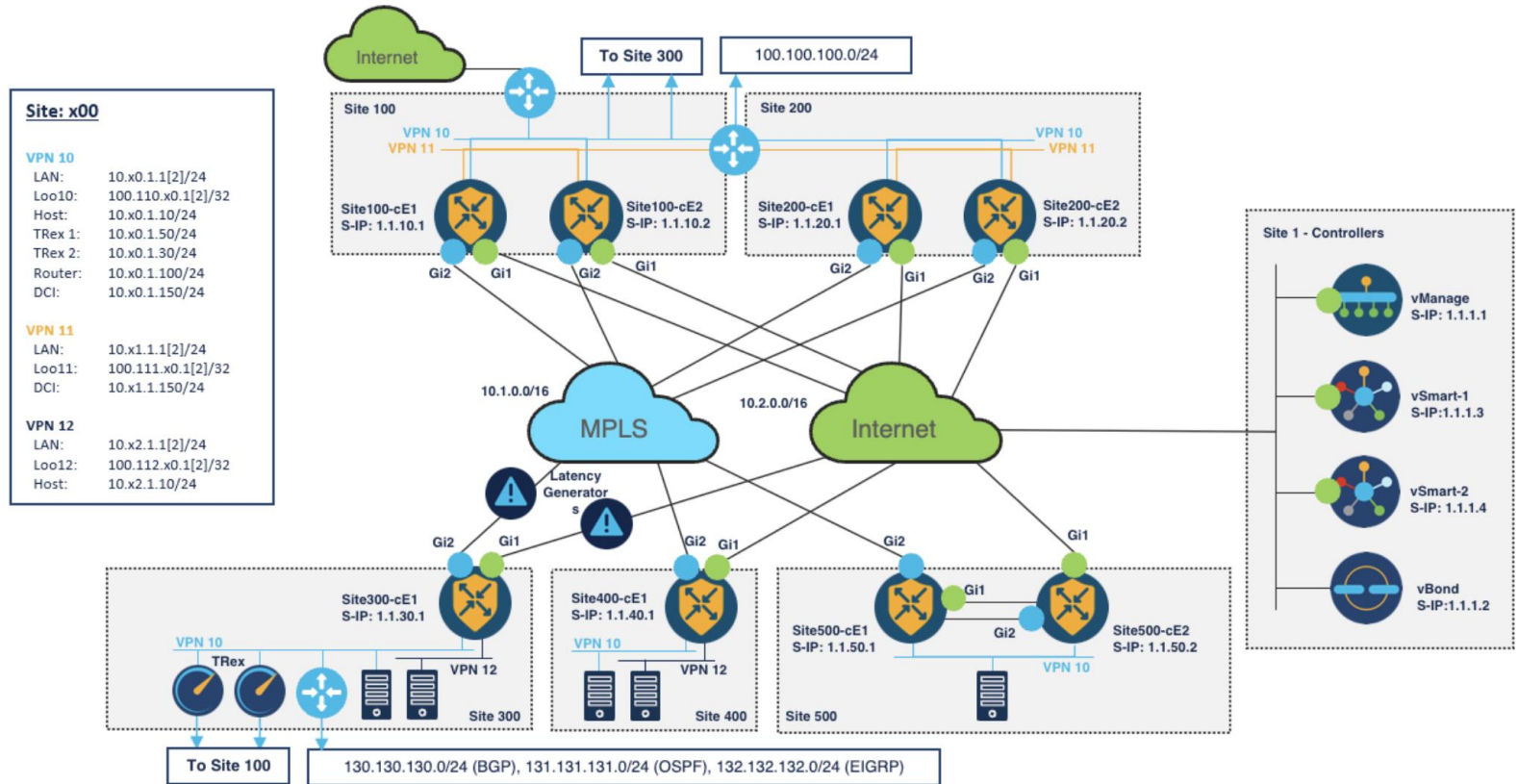
Demo Topology



Hub and Spoke Topology



Demo Topology



Quiz Time

SD-WAN

Learn how to confidently deploy and operate Cisco's SD-WAN solution in a new or existing network. These sessions provide a journey from the foundation to latest Cisco SD-WAN innovations focusing on design, innovations, and integrations with Cloud, SASE, and Assurance/Analytics.

Monday-Thursday

START

[LABENT-2547](#)

Troubleshooting Cisco SD-WAN Control Plane

[LABENT-2544](#)

Troubleshooting Cisco SD-WAN Data Plane

[LABENT-2490](#)

Understanding SD-WAN Overlay Management Protocol

[LABENT-2570](#)

Unified Security Policy in Cisco SD-WAN

[LABENT-2581](#)

Adaptive QoS in Cisco SD-WAN

Monday, February 5 | 8:30 a.m.

[TECENT-3377](#)

Deep dive into Cisco SD-WAN - how it works

Monday, February 5 | 8:45 a.m.

[TECENT-2208](#)

Implementing SD-WAN Branch on-prem/ SASE Security

Monday, February 5 | 2:00 p.m.

[TECENT-2376](#)

From Zero to SD-WAN Hero: Planning, Designing and Implementing Cisco SD-WAN

Tuesday, February 6 | 8:00 a.m.

[BRKENT-2108](#)

Cisco SD-WAN: Start Here

Tuesday, February 6 | 8:00 a.m.

[BRKENT-2524](#)

Multicloud Security Unleashed: Bridging the Gap Between SD-WAN, Clouds, Firewall Service Insertion, Valtix and Secure Internet Gateways

Tuesday, February 6 | 9:30 a.m.

[BRKENT-2660](#)

Customer Case Studies: Lessons learned from Cisco's SD-WAN Design Council

Tuesday, February 6 | 2:00 p.m.

[BRKENT-2043](#)

Harnessing the Capabilities of the Cisco Catalyst SD-WAN Policy Framework: Architecture, Building Blocks, and Case Studies.

Wednesday, February 7 | 8:45 a.m.

[BRKENT-3797](#)

Advanced SD-WAN Policies Troubleshooting

Wednesday, February 7 | 1:45 p.m.

[BRKENT-2283](#)

4 steps to unify Multi-Cloud Connectivity and Design with Cisco SD-WAN principles

Thursday, February 8 | 10:30 a.m.

[BRKTRTS-3050](#)

Cisco SD-WAN - Hidden Complexity Revealed - How Cisco TAC Addresses Really Tricky Problems?

Thursday, February 8 | 10:30 a.m.

[BRKENT-2006](#)

Optimizing and Orchestrating End Users' Connections to Public and Private Clouds in a SASE World

FINISH

If you are unable to attend a live session, you can watch it in the On-demand library.

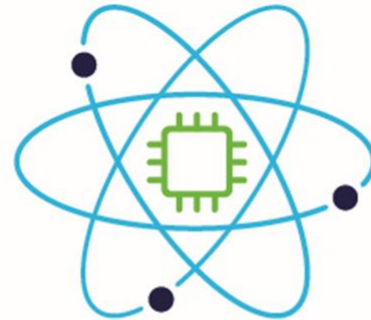
Become a power user

Cisco Live Amsterdam Exclusive:
Save 25% on CML-Personal and
CML-Personal Plus.

Visit the Learning & Certification
booth to learn more.



CISCO *Live!*



Powered by
Cisco Modeling Labs



The bridge to possible

Thank you

CISCO *Live!*

The Cisco Live! logo features the word "CISCO" in a bold, black, sans-serif font, followed by "Live!" in a black, cursive script font. The background of the entire image is a vibrant, multi-colored abstract pattern of overlapping, wavy bands in shades of red, orange, yellow, green, and blue, creating a sense of motion and energy.

CISCO *Live!*

Let's go