

Layer2 EVPN over Segment Routing MPLS

- About Layer 2 EVPN, on page 1
- Guidelines and Limitations for Layer 2 EVPN over Segment Routing MPLS, on page 2
- Configuring Layer 2 EVPN over Segment Routing MPLS, on page 2
- Configuring VLAN for EVI, on page 6
- Configuring the NVE Interface, on page 6
- Configuring EVI Under VRF, on page 7
- Configuring Anycast Gateway, on page 7
- Advertising Labelled Path for the Loopback Interface, on page 7
- About SRv6 Static Per-Prefix TE, on page 8
- Configuring a SRv6 Static Per-Prefix TE, on page 8
- About Route-Target Auto, on page 11
- Configuring RD and Route Targets for BD, on page 12
- Configuring RD and Route Targets for VRF, on page 12
- Configuration Examples for Layer 2 EVPN over Segment Routing MPLS, on page 13

About Layer 2 EVPN

Ethernet VPN (EVPN) is a next generation solution that provides ethernet multipoint services over MPLS networks. EVPN operates in contrast to the existing Virtual Private LAN Service (VPLS) by enabling control-plane based MAC learning in the core. In EVPN, PEs participating in the EVPN instances learn customer MAC routes in control-plane using MP-BGP protocol. Control-plane MAC learning brings several benefits that allow EVPN to address the VPLS shortcomings, including support for multihoming with per-flow load balancing.

In a data center network, the EVPN control plane provides:

- Flexible workload placement that is not restricted with the physical topology of the data center network. Therefore, you can place virtual machines (VM) anywhere within the data center fabric.
- Optimal East-West traffic between servers within and across data centers. East-West traffic between servers, or virtual machines, is achieved by most specific routing at the first hop router. First hop routing is done at the access layer. Host routes must be exchanged to ensure most specific routing to and from servers or hosts. VM mobility is supported by detecting new endpoint attachment when a new MAC address or the IP address is directly connected to the local switch. When the local switch sees the new MAC or the IP address, it signals the new location to rest of the network.

- Segmentation of Layer 2 and Layer 3 traffic, where traffic segmentation is achieved using MPLS encapsulation and the labels (per-BD label and per-VRF labels) act as the segment identifier.

Guidelines and Limitations for Layer 2 EVPN over Segment Routing MPLS

Layer 2 EVPN over segment routing MPLS has the following guidelines and limitations:

- Segment routing Layer 2 EVPN flooding is based on the ingress replication mechanism. MPLS core does not support multicast.
- ARP suppression is not supported.
- Non-IPv4/IPv6 traffic with source MAC addresses beginning with values 4 or 6 (for example, 4x:xx:xx:xx:xx:xx or 6x:xx:xx:xx:xx:xx) may be dropped when sent over Layer 2 EVPN tunnels. This limitation affects non-IP traffic such as ARP and IPX. While IP traffic is not directly affected, ARP resolution failures may impact IP host reachability.

  This limitation can be overcome by inserting a control word (CW) in the MPLS encapsulation using the control-word option in evpn encapsulation mpls configuration.

  Example:

```
switch# configure terminal
switch(config)# evpn
switch(config-evpn)# evi 100
switch(config-evpn-evi)# encapsulation mpls
switch(config-evpn-evi-mpls)# source-interface loopback0
switch(config-evpn-evi-mpls)# control-word
```

- Consistency checking on vPC is not supported.
- The same Layer 2 EVI and Layer 3 EVI cannot be configured together.
- Beginning with Cisco NX-OS Release 9.3(1), Layer 2 EVPN is supported on Cisco Nexus 9300-FX2 platform switches.
- Beginning with Cisco NX-OS Release 9.3(5), Layer 2 EVPN over segment routing MPLS is supported on Cisco Nexus 9300-GX and Cisco Nexus 9300-FX3 platform switches.

Configuring Layer 2 EVPN over Segment Routing MPLS

Before you begin

Do the following:

- You must install and enable the MPLS feature set using the **install feature-set mpls** and **feature-set mpls** commands.
- You must enable the MPLS segment routing feature.
- You must enable the nv overlay feature using the **nv overlay** command.

- You must enable EVPN control plane using the **nv overlay evpn** command.

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**feature bgp** **Example:** `switch(config)#feature bgp`	Enables BGP feature and configurations.
Step 3	**install feature-set mpls** **Example:** `switch(config)#install feature-set mpls`	Enables MPLS configuration commands.
Step 4	**feature-set mpls** **Example:** `switch(config)#install feature-set mpls`	Enables MPLS configuration commands.
Step 5	**feature mpls segment-routing** **Example:** `switch(config)#feature mpls segment-routing`	Enables segment routing configuration commands.
Step 6	**feature mpls evpn** **Example:** `switch(config)#feature mpls evpn`	Enables EVPN over MPLS configuration commands. This command is mutually exclusive with the **feature-nv** CLI command.
Step 7	**feature nv overlay** **Example:** `switch(config)#feature nv overlay`	Enables the NVE feature that is used for the segment routing Layer 2 EVPN.
Step 8	**nv overlay evpn** **Example:** `switch(config)#nv overlay evpn`	Enables EVPN.
Step 9	**interface loopback** *Interface_Number* **Example:** `switch(config)#interface loopback 1`	Configures the loopback interface for NVE.
Step 10	**ip address** *address* **Example:**	Configures the IP address.

	Command or Action	Purpose
	`switch(config-if)#ip address 192.168.15.1`	
Step 11	**exit** **Example:** `switch(config-if)#exit`	Exits global address family configuration mode.
Step 12	**evpn** **Example:** `switch(config)#evpn`	Enters the EVPN configuration mode.

Step 13	**evi** *number*
	Example:
	`switch(config-evpn)#evi 1000`
	`switch(config-evpn-sr)#`
		Configures Layer 2 EVI. If required, you can manually configure the RT based on the EVI that is generated automatically.

Step 14	**encapsulation mpls** **Example:** `switch(config-evpn)#encapsulation mpls`	Enables MPLS encapsulation and ingress-replication.
Step 15	**source-interface loopback** *Interface_Number* **Example:** `switch(config-evpn-nve-encap)#source-interface loopback 1`	Specifies the NVE source interface.
Step 16	**exit** **Example:** `switch(config-evpn-nve-encap)#exit`	Exits the configuration.
Step 17	**vrf context** *VRF_NAME* **Example:** `switch(config)#vrf context Tenant-A`	Configures the VRF.
Step 18	**evi** *EVI_ID* **Example:** `switch(config-vrf)#evi 30001`	Configures L3 EVI.
Step 19	**exit** **Example:** `switch(config-vrf)#exit`	Exits the configuration.
Step 20	**VLAN** *VLAN_ID* **Example:** `switch(config)#vlan 1001`	Configures VLAN.

	Command or Action	Purpose
Step 21	**evi auto** **Example:** `switch(config-vlan)#evi auto`	Configures L2 EVI.
Step 22	**exit** **Example:** `switch(config-vlan)#exit`	
Step 23	**router bgp** *autonomous-system-number* **Example:** `switch(config)#router bgp 1`	Enters the BGP configuration mode.
Step 24	**address-family l2vpn evpn** **Example:** `switch(config-router)#address-family l2vpn evpn`	Enables EVPN address family globally.
Step 25	**neighbor address** *remote-as autonomous-system-number* **Example:** `switch(config-router)#neighbor 192.169.13.1 remote as 2`	Configures BGP neighbor.
Step 26	**address-family l2vpn evpn** **Example:** `switch(config-router-neighbor)#address-family l2vpn evpn`	Enables EVPN address family for neighbor.
Step 27	**encapsulation mpls** **Example:** `switch(config-router-neighbor)#encapsulation mpls`	Enables MPLS encapsulation.
Step 28	**send-community extended** **Example:** `switch(config-router-neighbor)#send-community extended`	Configures BGP to advertise extended community lists.
Step 29	**vrf** *VRF_NAME* **Example:** `switch(config-router)#vrf Tenant-A`	Configures BGP VRF.
Step 30	**exit** **Example:** `switch(config-router)#exit`	Exits the configuration.

Configuring VLAN for EVI

Procedure

	Command or Action	Purpose
Step 1	**vlan** *number*	Specifies the VLAN.
Step 2	**evi** *auto*	Creates a BD label for the VLAN. This label is used as an identifier for the VLAN across the segment routing Layer 2 EVPN.

Configuring the NVE Interface

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**interface loopback** *loopback_number* **Example:** `switch(config)# interface loopback 1`	Associates the IP address with this loopback interface and uses this IP address for the segment routing configuration.
Step 3	**ip address** **Example:** `switch(config-if)#ip address 192.169.15.1/32`	Specifies the IPv4 address family and enters router address family configuration mode.
Step 4	**evpn** **Example:** `switch(config)#evpn`	Enters EVPN configuration mode.
Step 5	**encapsulation mpls** **Example:** `switch(config-evpn)# encapsulation mpls`	Enables MPLS encapsulation and ingress-replication.
Step 6	**source-interface** *loopback_number* **Example:** `switch(config-evpn-nve-encap)#source-interface loopback 1`	Specifies the NVE source interface.

	Command or Action	Purpose
Step 7	**exit** **Example:** `switch(config)# exit`	Exits segment routing mode and returns to the configuration terminal mode.

Configuring EVI Under VRF

Procedure

	Command or Action	Purpose
Step 1	**vrf context** *tenant*	Create a VRF Tenant.
Step 2	**evi** *number*	Configure Layer 3 EVI under VRF.

Configuring Anycast Gateway

The fabric forwarding configuration is necessary only if the SVIs are configured in the anycast mode.

Procedure

	Command or Action	Purpose
Step 1	**fabric forwarding anycast-gateway-mac 0000.aabb.ccdd**	Configures the distributed gateway virtual MAC address.
Step 2	**fabric forwarding mode anycast-gateway**	Associates SVI with the Anycast Gateway under the interface configuration mode.

Advertising Labelled Path for the Loopback Interface

The loopback interface, advertised as Layer 2 EVPN endpoint should be mapped to a label index. Thereby BGP advertises MPLS labelled path for the same.

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

	Command or Action	Purpose
Step 2	[**no**]**router ospf** *process* **Example:** `switch(config)# router ospf test`	Enables the OSPF mode.
Step 3	**segment-routing** **Example:** `switch(config-router)# segment-routing mpls`	Configures the segment routing functionality under OSPF.
Step 4	**connected-prefix-sid-map** **Example:** `switch(config-sr-mpls)# connected-prefix-sid-map`	Enters a sub-mode where you can configure address-family specific mappings for local prefixes and SIDs.
Step 5	**address-family ipv4** **Example:** `switch(config-sr-mpls-conn)# address-family ipv4`	Specifies IPv4 address prefixes.
Step 6	**1.1.1.1/32 index 100** **Example:** `switch(config-sr-mpls-conn-af)# 1.1.1.1/32 100`	Associates SID 100 with the address 1.1.1.1/32.
Step 7	**exit-address-family** **Example:** `switch(config-sr-mpls-conn-af)# exit-address-family`	Exits the address family.

About SRv6 Static Per-Prefix TE

The SRv6 Static Per-Prefix TE feature allows you to map and advertise prefixes that at mapped to non-default VRFs. This feature allows you to advertise multiple prefixes in a single instance using the matching VRF route target and prevents the manual entry of each prefix.

In Cisco NX-OS Release 9.3(5), only one VNF can service a VM.

Configuring a SRv6 Static Per-Prefix TE

Before you begin

Do the following:

- You must install and enable the MPLS feature set using the **install feature-set mpls** and **feature-set mpls** commands.

- You must enable the MPLS segment routing feature.

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**vrf context** *VRF_Name* **Example:** `switch(config)# vrf context vrf_2_7_8`	Defines VRF and enters the VRF configuration mode.
Step 3	**rd** *rd_format* **Example:** `switch(config-vrf)# rd 2.2.2.0:2`	Assign the RD to VRF.
Step 4	**address-family** {**ipv4** | **ipv6** } **Example:** `switch(config-vrf)# address-family ipv4 unicast`	Specifies either the IPv4 or the IPv6 address family for the VRF instance and enters the address family configuration mode.
Step 5	**route-target import** *route-target-id* **Example:** `switch(config-vrf)# route-target import 1:2`	Configures the importing of routes to the VRF.
Step 6	**route-target import** *route-target-id* **evpn** **Example:** `switch(config-vrf)# route-target import 1:2 evpn`	Configures importing of routes that have a matching route target value from the Layer 3 EVPN to the VRF.
Step 7	**route-target export** *route-target-id* **Example:** `switch(config-vrf)# route-target export 1:2`	Configures the exporting of routes from the VRF.
Step 8	**route-target export** *route-target-id* **evpn** **Example:** `switch(config-vrf)# route-target export 1:2 evpn`	Configures exporting of routes that have a matching route target value from the VRF to the Layer 3 EVPN.
Step 9	**router bgp** *autonomous-system-number* **Example:** `switch(config)# router bgp 65000`	Enables BGP and assigns the AS number to the local BGP speaker.

	Command or Action	Purpose
Step 10	**router-id** *id* **Example:** `switch(config-router)# router-id 2.2.2.0`	Configures the router ID.
Step 11	**address-family l2vpn evpn** **Example:** `switch(config-router-af)# address-family l2vpn evpn`	Enters global address family configuration mode for the Layer 2 VPN EVPN.

Step 12	**neighbor** *ipv4-address* **remote-as**
	Example:
	`switch(config-router)# neighbor 7.7.7.0 remote-as 65000`
	`switch(config-router-neighbor)#`
		Configures the IPv4 address and AS number for a remote BGP peer.

Step 13	**update-source loopback** *number* **Example:** `switch(config-router-neighbor)# update-source loopback0`	Specifies the loopback number.
Step 14	**address-family l2vpn evpn** **Example:** `switch(config-router-neighbor)#address-family l2vpn evpn`	Enables EVPN address family for a neighbor.
Step 15	**send-community extended** **Example:** `switch(config-router-neighbor)#send-community extended`	Configures BGP to advertise extended community lists.
Step 16	**encapsulation mpls** **Example:** `switch(config-router-neighbor)#encapsulation mpls`	Enables MPLS encapsulation.
Step 17	**exit** **Example:** `switch(config-router-neighbor)#exit`	Exits the configuration.

Example

The following example shows how to configure RPM configuration in order to define the VRF VT.

```
rf context vrf_2_7_8
          rd 2.2.2.0:2
          address-family ipv4 unicast
            route-target import 0.0.1.1:2
            route-target import 0.0.1.1:2 evpn
```

```
            route-target export 0.0.1.1:2
            route-target export 0.0.1.1:2 evpn
ip extcommunity-list standard vrf_2_7_8-test permit rt 0.0.1.1:2
            route-map Node-2 permit 4
            match extcommunity vrf_2_7_8-test
            set extcommunity color 204
```

About Route-Target Auto

The auto-derived Route-Target (route-target import/export/both auto) is based on the Type 0 encoding format as described in IETF RFC 4364 section 4.2 (https://tools.ietf.org/html/rfc4364#section-4.2). IETF RFC 4364 section 4.2 describes the Route Distinguisher format and IETF RFC 4364 section 4.3.1 refers that it is desirable to use a similar format for the Route-Targets. The Type 0 encoding allows a 2-byte administrative field and a 4-byte numbering field. Within Cisco NX-OS, the auto derived Route-Target is constructed with the Autonomous System Number (ASN) as the 2-byte administrative field and the Service Identifier (EVI) for the 4-byte numbering field.

2-byte ASN

The Type 0 encoding allows a 2-byte administrative field and a 4-byte numbering field. Within Cisco NX-OS, the auto-derived Route-Target is constructed with the Autonomous System Number (ASN) as the 2-byte administrative filed and the Service Identifier (EVI) for the 4-byte numbering field.

Examples of an auto derived Route-Target (RT):

- IP-VRF within ASN 65001 and L3EVI 50001 - Route-Target 65001:50001
- MAC-VRF within ASN 65001 and L2EVI 30001 - Route-Target 65001:30001

For Multi-AS environments, the Route-Targets must either be statically defined or rewritten to match the ASN portion of the Route-Targets.

Note Auto derived Route-Targets for a 4-byte ASN are not supported.

4-byte ASN

The Type 0 encoding allows a 2-byte administrative field and a 4-byte numbering field. Within Cisco NX-OS, the auto-derived Route-Target is constructed with the Autonomous System Number (ASN) as the 2-byte administrative filed and the Service Identifier (EVI) for the 4-byte numbering field. With the ASN demand of 4-byte length and the EVI requiring 24-bit (3-bytes), the Sub-Field length within the Extended Community is exhausted (2-byte Type and 6-byte Sub-Field). As a result of the length and format constraint and the importance of the Service Identifiers (EVI) uniqueness, the 4-byte ASN is represented in a 2-byte ASN named AS_TRANS, as described in IETF RFC 6793 section 9 (https://tools.ietf.org/html/rfc6793#section-9). The 2-byte ASN 23456 is registered by the IANA (https://www.iana.org/assignments/iana-as-numbers-special-registry/iana-as-numbers-special-registry.xhtml) as AS_TRANS, a special purpose AS number that aliases 4-byte ASNs.

Example auto derived Route-Target (RT) with 4-byte ASN (AS_TRANS):

- IP-VRF within ASN 65656 and L3EVI 50001 - Route-Target 23456:50001
- MAC-VRF within ASN 65656 and L2EVI 30001 - Route-Target 23456:30001

Configuring RD and Route Targets for BD

The Bridge Domain (BD) RD and Route Targets are automatically generated when you configure **evi auto** under the VLAN. To configure the BD RD and Route Targets manually, perform these steps:

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**evpn** **Example:** `switch(config)# evpn`	Enters EVPN configuration mode.
Step 3	**evi** *VLAN_ID* **Example:** `switch(config-evpn)# evi 1001`	Specifies L2 EVI to configure RD/Route Target.
Step 4	**rd** *rd_format* **Example:** `switch(config-evpn-evi-sr)# rd 192.1.1.1:33768`	Configures RD.
Step 5	**route-target both** *rt_format* **Example:** `switch(config-evpn-evi-sr)# route-target both 1:20001`	Configures Route Target.

Configuring RD and Route Targets for VRF

The VRF RD and Route Targets are automatically generated when you configure the **evi** *evi_ID* under the VRF. To configure the VRF RD and Route Targets manually, perform these steps:

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

	Command or Action	Purpose
Step 2	**vrf context** *VRF_NAME* **Example:** `switch(config)# vrf context A`	Configures the VRF.
Step 3	**rd auto** or **rd_format** **Example:** `switch(config-vrf)# rd auto`	Configures RD.
Step 4	**address-family ipv4 unicast** **Example:** `switch(config-vrf)# address-family ipv4 unicast`	Enables IPv4 address family.
Step 5	**route-target both** *rt_format* **evpn** **Example:** `switch(config-vrf-af-ipv4)# route-target both 1:30001 evpn`	Configures Route Target.

Configuration Examples for Layer 2 EVPN over Segment Routing MPLS

The following examples show the configuration for Layer 2 EVPN over Segment Routing MPLS:

```
install feature-set mpls
feature-set mpls
nv overlay evpn
feature bgp
feature mpls segment-routing
feature mpls evpn
feature interface-vlan
feature nv overlay

fabric forwarding anycast-gateway-mac 0000.1111.2222

vlan 1001
  evi auto

vrf context Tenant-A
  evi 30001

interface loopback 1
  ip address 192.168.15.1/32

interface vlan 1001
  no shutdown
  vrf member Tenant-A
  ip address 111.1.0.1/16
  fabric forwarding mode anycast-gateway

router bgp 1
  address-family l2vpn evpn
```

```
    neighbor 192.169.13.1
      remote-as 2
      address-family l2vpn evpn
        send-community extended
        encapsulation mpls
    vrf Tenant-A

evpn
  encapsulation mpls
    source-interface loopback 1
```