Trunk EFP Support

The Trunk EFP Support feature provides support for Ethernet flow points (EFPs) on trunk ports. A trunk port allows a range of VLANs to be forwarded on a given interface while still maintaining data-plane segmentation between the VLANs.

Restrictions for Trunk EFP Support

- The **rewrite ingress tag pop 1 symmetric** command is the only **rewrite** command that is supported for trunk EFP configurations. The **rewrite ingress tag pop 1 symmetric** command must be included in the configuration when the Trunk EFP Support feature is enabled.
- A bridge-domain number that is part of a trunk EFP configuration cannot be shared by other EFPs under the same port or interface.
- Only one trunk EFP can be configured under one port or interface.
- All features configured on a trunk EFP (other than encapsulations and bridge-domain assignments) are applied uniformly to all VLANs and bridge domains. If a feature requires VLAN-specific or bridge-domain-specific configuration values, the feature cannot be applied on the trunk EFP. Those special VLANs or bridge domains must be removed from the EFP trunk to form individual EFPs.
- Trunk EFP MET supports a maximum of 4078 VLANs and the maximum threshold supported is 20480.
- Untagged EFP should be added to the BDI when untagged packets are directed towards the interface to avoid packets punting to host queue.

RSP3 Module:

- L2 port will start dropping untagged traffic when untagged/default/ptag EFP is not configured and it may start impacting any control plane protocol which requires untagged traffic to be processed. If that happens, you need to explicitly configure the untagged EFP. For example, LACP.

Restrictions for Trunk EFP Support

- Only 1000 VLANs can be configured for a trunk EFP (with or without port channel).
- Trunk EVC and encapsulation default EVC cannot co-exist on the same interface.
- Dynamically changing the Trunk EFP number on an interface is *not* supported on the RSP3 module.

Information About Trunk EFP Support

Benefits of Trunk EFP Support

The Carrier Ethernet infrastructure supports the following types of Ethernet flow points (EFPs):

- Static EFPs that are user-configurable.
- Dynamic EFPs that are created and maintained during a Cisco Intelligent Services Gateway (ISG) session.

With this feature, a new EFP type has been added that is intended for use on a trunk port.

A trunk port allows a range of VLANs to be forwarded on a given interface while maintaining data-plane segmentation between the VLANs.

Note Trunk EFP (with or without port channel) supports encapsulation of up to 1000 VLANs.

Like a static EFP, this new type of EFP is user-configurable via the **service instance trunk** command, the **encapsulation** command, and the **bridge-domain from-encapsulation** command when the Trunk EFP Support feature is enabled.

Ethernet Flow Points

An Ethernet flow point (EFP) is a forwarding decision point in the provider edge (PE) router, which gives network designers flexibility to make many Layer 2 flow decisions within the interface. Many EFPs can be configured on a single physical port. (The number varies from one device to another.) EFPs are the logical demarcation points of an Ethernet virtual connection (EVC) on an interface. An EVC that uses two or more user network interfaces (UNIs) requires an EFP on the associated ingress and egress interfaces of every device that the EVC passes through.

EFPs can be configured on any Layer 2 traffic port; however, they are usually configured on UNI ports. The following parameters (matching criteria) can be configured on the EFP:

- Frames of a specific VLAN, a VLAN range, or a list of VLANs (100-150 or 100,103,110)
- Frames with no tags (untagged)
- Frames with identical double-tags (VLAN tags) as specified
- Frames with identical Class of Service (CoS) values

A frame passes each configured match criterion until the correct matching point is found. If a frame does not fit any of the matching criteria, it is dropped. Default criteria can be configured to avoid dropping frames.

You can configure a new type of TEFP called TEFP with encapsulation from bridge domain (BD). All the BDs configured on the switch are part of the VLAN list of the encapsulated TEFP. The TEFP is encapsulated using the **encapsulation dot1q from-bd** command. The feature brings about the following interaction between the Ethernet-EFP and Layer2-bridge domain components:

- If BDs exist in the system and a TEFP with encapsulation from bridge domain is created, then all the BDs get added to the VLAN list of TEFP with encapsulation from bridge domain.
- If TEFP with encapsulation from bridge domain exists in the system and a new BD is created, then the BD is added to the VLAN list of all the TEFP with encapsulation from bridge domain in the system.
- If TEFP with encapsulation from bridge domain exists in the system and a BD gets deleted, and if the deleted BD is not part of an existing TEFP or EFP then it gets deleted from all the TEFP with encapsulation from bridge domain in the system.

The following types of commands can be used in an EFP:

- Rewrite commands—In each EFP, VLAN tag management can be specified with the following actions:
  - Pop—1) pops out a tag; 2) pops out two tags
  - Push— pushes in a tag
  - Translate—1 to 1) changes a tag value; 1 to 2) pops one tag and pushes two tags; 2 to 1) pops two tags and pushes one tag; 2 to 2) changes the value for two tags
- Forwarding commands—Each EFP specifies the forwarding command for the frames that enter the EFP. Only one forwarding command can be configured per EFP. The forwarding options are as follows:
  - Layer 2 point-to-point forwarding to a pseudowire tunnel
  - Multipoint bridge forwarding to a bridge domain entity
  - Local switch-to-switch forwarding between two different interfaces
- Feature commands—In each EFP, the QoS features or parameters can be changed and the ACL can be updated.

Trunk Ports

An Ethernet interface can be configured as a trunk port (interface). A trunk port, also known as a trunk, is a point-to-point link between a networking device and another networking device. Trunks carry the traffic of multiple VLANs over a single link and allow you to extend VLANs across an entire network. A trunk port configured on the interface with two or more VLANs can carry traffic for several VLANs simultaneously.

To correctly deliver the traffic on a trunk port with several VLANs, the device uses the IEEE 802.1Q encapsulation or tagging method.

How to Enable Trunk EFP Support

Enabling Trunk EFP Support

To enable Ethernet flow point (EFP) support on a trunk port or trunk interface, complete the following steps.

Note TEFP is supported on a PC interface and on a Gigabit interface. The procedure listed below is for TEFP configuration on a PC interface. Similar procedure is used for TEFP configuration on a gigabit interface.

Note When configuring TEFP on a port-channel interface, ensure that the port interface is always up.

Procedure

Step 1 **enable**

Example:

```
Device> enable
```

Enables privileged EXEC mode.

- Enter your password if prompted.

Step 2 **configure terminal**

Example:

```
Device# configure terminal
```

Enters global configuration mode.

Step 3 **interface** *port-channel number*

Example:

```
Device(config)# interface port-channel 1
```

Configures the interface and enters interface configuration mode.

Step 4 **service instance trunk** *id* **ethernet**

Example:

```
Device(config-if)# service instance trunk 1 ethernet
```

Configures an Ethernet service instance on an interface and enters Ethernet service configuration mode.

Step 5 **encapsulation dot1q** {**from-bd** |*vlan-id* [**,** *vlan-id* [**-** *vlan-d*]]}

Example:

```
Device(config-if-srv)# encapsulation dot1q 1-5, 7, 9-12
```

```
Device(config-if-srv)# encapsulation dot1q from-bd
```

Defines the matching criteria to map 802.1Q frames ingress on an interface to the appropriate service instance.

Step 6 **rewrite ingress tag pop 1 symmetric**

Example:

```
Device(config-if-srv)# rewrite ingress tag pop 1 symmetric
```

Specifies the encapsulation adjustment to be performed on a frame that is entering a service instance.

Step 7 **bridge-domain from-encapsulation**

Example:

```
Device(config-if-srv)# bridge-domain from-encapsulation
```

Creates a list of bridge domains for an EFP trunk port using the bridge-domain IDs derived from the encapsulation VLAN numbers.

Step 8 **no shutdown**

Example:

```
Device(config-if-srv)# no shutdown
```

Disables shutdown and keeps the interface or port active.

Step 9 **end**

Example:

```
Device(config-if-srv)# end
```

Returns to privileged EXEC mode.

Verifying the Trunk EFP Support Configuration

Use one or more of the commands listed below to verify the Trunk EFP Support feature configuration.

Procedure

Step 1 **enable**

Example:

```
Device> enable
```

Enables privileged EXEC mode.

- Enter your password if prompted.

Step 2 **show ethernet service instance**

Example:

```
Device# show ethernet service instance
```

Displays information about Ethernet service instances.

Step 3 **show ethernet service instance interface port-channel** [*number*]

Example:

```
Device# show ethernet service instance interface port-channel 1
```

Displays interface-only information about Ethernet service instances for all port-channel interfaces or for a specified port-channel interface.

Step 4 **show bridge-domain**

Example:

```
Device# show bridge-domain
```

Displays bridge-domain information.

Step 5 **exit**

Example:

```
Device# exit
```

Exits privileged EXEC mode.

Configuration Examples

Example: Configuring Trunk EFP Support

In the following example, EFP support has been configured on a trunk interface.

```
Device> enable
Device# configure terminal
Device(config)# interface port-channel 1
Device(config-if)# service instance trunk 1 ethernet
Device(config-if-srv)# encapsulation dot1q 1 - 5, 7, 9 - 12
Device(config-if-srv)# rewrite ingress tag pop 1 symmetric
Device(config-if-srv)# bridge-domain from-encapsulation
Device(config-if-srv)# no shutdown
Device(config-if-srv)# end
```

Example: Verifying the Trunk EFP Support Configuration

The following is sample output from the **show ethernet service instance** command. The output displays trunk as the service instance type and indicates that a bridge domain for VLANs in the range of 12 to 1900 (as specified by the encapsulation parameters) has been created for service instance 4000 on a trunk port (interface).

```
Device# show ethernet service instance id 4000 interface port-channel 1

Service Instance ID: 4000
Service Instance Type: Trunk
Associated Interface Port-channel: 1
Associated EVC:
L2protocol drop
CE-Vlans:
Encapsulation: dot1q 12-1900 vlan protocol type 0x8100
Rewrite: ingress tag pop 1 symmetric
Interface Port-channel Dot1q Tunnel Ethertype: 0x8100
State: Up
EFP Statistics:
   Pkts In   Bytes In   Pkts Out  Bytes Out
168729725 10798985220  160246675 10255787200
EFP Microblocks:
****************
Microblock type: Bridge-domain
Bridge-domain: 12-1900
```