

Ethernet Virtual Connections on Port Channels

A port channel bundles individual Ethernet links into a single logical link that provides an aggregated bandwidth of up to four physical links. The Ethernet Virtual Connection (EVC) Port Channel feature provides support for the Ethernet service instances on a port channel.

- Information About Ethernet Virtual Connections on Port Channels, page 1
- Configuring a Basic EVC Port Channel, page 2
- Load Balancing the EVCs on Port Channels, page 6
- , page 7
- Configuring Flow Based Load Balancing, page 8
- VLAN-Based Manual Load Balancing, page 9
- Configuring LACP, page 14

Information About Ethernet Virtual Connections on Port Channels

Usage Guidelines for Configuring an EVC on a Port Channel

Prior to configuring an EVC on a port channel, consider these usage guidelines:

- All the member links of the port channel are on the Cisco ASR 1000 Fixed Ethernet Line card or on the shared port adapters (SPAs).
- All the member links of the port channel must be configured such that they are of the same speed, and are in the same duplex mode.
- EVC connect and IP subinterfaces are allowed to co-exist over the port-channel interface.
- If you configure a physical port as part of a channel group, you cannot configure EVCs under that physical port.
- You cannot use the bandwidth percent or police percent commands on EVC port channels in flat policy maps or in the parent of the Hierarchical quality of service (HQoS) policy maps.

Quality of Service Support

The following Quality of Service (QoS) policy settings are supported for egress traffic on the Cisco ASR 1000 Layer 3 port-channel interface:

- The QoS on the subinterface of a port channel with a specific encapsulation, such as, VLAN.
- The QoS on a member link.
- Modular QoS CLI (MQC) policy aggregation is supported with VLAN-based load balancing on subinterfaces and member-link ports.

The following section describes the QoS support for the following interfaces:

Port-Channel Interface

- Does not support any QoS policies on the port-channel interface.

Member-Link Interface

- Supports egress queuing, policing, and marking for flow-based load balancing.
- Supports egress queuing, policing, and marking for VLAN-based manual load balancing.
- Supports egress queuing, policing, marking, and the configuration of similar policies on two links for LACP 1:1.

EVC on Port-Channel Interface

- Does not support flow-based load balancing.
- Supports ingress and egress policing and marking for VLAN-based manual load balancing.
- Supports ingress and egress policing and marking for LACP 1:1.



Note Ensure that you configure the QoS policies for the EVCs on a port channel and a member link separately.



Note Service policies for ingress traffic are supported on EVC only.

Configuring a Basic EVC Port Channel

Configuring a Port Channel Interface with ethernet Service Instances

To configure a port-channel interface with Ethernet service instances, perform the following steps.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface port-channel** *number*
4. [**no**] **ip address**
5. **no negotiation auto**
6. [**no**] **service instance** *id* **Ethernet** [*service-name*]
7. **encapsulation** { **default** | **untagged** | **dot1q** *vlan-id* [**second-dot1q** *vlan-id*] }
8. [**no**] **service instance** *id* **ethernet** [*service-name*]
9. **encapsulation** {**default** | **untagged** | **dot1q** *vlan-id* [**second-dot1q** *vlan-id*] }

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router# enable`	Enables the privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters the global configuration mode.
Step 3	**interface port-channel** *number* **Example:** `Router(config)# interface port-channel 11`	Creates the port-channel interface.
Step 4	[**no**] **ip address** **Example:** `Router(config-if)# no ip address`	Removes an IP address or disables IP processing.
Step 5	**no negotiation auto** **Example:** `Router(config-if)# no negotiation auto`	Disables the advertisement of speed, duplex mode, and flow control on a Gigabit Ethernet interface.
Step 6	[**no**] **service instance** *id* **Ethernet** [*service-name*] **Example:** `Router(config-if)# service instance 1 ethernet`	Creates a service instance (an instantiation of an EVC) on an interface and sets the device to the config-if-srv submode.
Step 7	**encapsulation** { **default** | **untagged** | **dot1q** *vlan-id* [**second-dot1q** *vlan-id*] } **Example:** `Router(config-if-srv)# encapsulation dot1q 101`	Defines the matching criteria to be used in order to map ingress dot1q frames on an interface to the appropriate service instance.

	Command or Action	Purpose
Step 8	[**no**] **service instance** *id* **ethernet** [*service-name*] **Example:** `Router(config-if)# service instance 2 ethernet`	Creates a second service instance on the interface.
Step 9	**encapsulation** {**default** | **untagged** | **dot1q** *vlan-id* [**second-dot1q** *vlan-id*] } **Example:** `Router(config-if-srv)# encapsulation dot1q 201`	Defines the matching criteria to be used in order to map ingress dot1q frames on an interface to the appropriate service instance.

Adding a Port Channel Ethernet Flow Point to a Bridge Domain

To add a port channel Ethernet Flow Point (EFP) to a bridge domain, perform the following steps.

SUMMARY STEPS

1. [**no**] **bridge-domain** *bridge-id*
2. **member port-channel interface** *id* **service-instance** *id*
3. [**no**] **bridge-domain** *bridge-id* | **xconnect vfi** *vfi name*
4. **member port-channel interface** *id* **service-instance** *id*

DETAILED STEPS

	Command or Action	Purpose
Step 1	[**no**] **bridge-domain** *bridge-id* **Example:** `Router(config-if-srv)# bridge-domain 100`	The **bridge-domain** command binds the service instance to a bridge domain instance, where *bridge-id* is the identifier of the bridge-domain instance. To disable bridging, use the **no** form of this command.
Step 2	**member port-channel interface** *id* **service-instance** *id* **Example:** `Router(config-if-srv)# member Port-channel1 service-instance 1`	Creates a service instance on the interface.
Step 3	[**no**] **bridge-domain** *bridge-id* | **xconnect vfi** *vfi name* **Example:** `Router(config-if-srv)# bridge-domain 200`	The **bridge-domain** command binds the service instance to a bridge-domain instance, where *bridge-id* is the identifier of the bridge-domain instance. The **xconnect vfi** command specifies the Layer 2 virtual forwarding interface (VFI) that you are binding to the interface, where *vfi name* is the name of the VFI.

	Command or Action	Purpose
Step 4	**member port-channel interface** *id* **service-instance** *id* **Example:** `Router(config-if-srv)# member Port-channel1 service-instance 2`	Creates a service instance on the interface.

Adding an Ethernet Port to the Port Channel Interface

To configure the channel group number on the Ethernet port and to add an Ethernet port to the port-channel interface, perform the following steps.

SUMMARY STEPS

1. **interface GigabitEthernet** *slot*/*port*/*sub-port*
2. **no ip address**
3. **no negotiation auto**
4. **channel-group** *channel-group-number*
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**interface GigabitEthernet** *slot*/*port*/*sub-port* **Example:** `Router(config-if-srv)# interface GigabitEthernet0/2/0`	Specifies the Gigabit Ethernet port that is to be configured. This is the port in which the slot, port, or subport specifies the location of the Gigabit Ethernet port.
Step 2	**no ip address** **Example:** `Router(config-if-srv)# no ip address`	Removes an IP address or disables IP processing.
Step 3	**no negotiation auto** **Example:** `Router(config-if-srv)# no negotiation auto`	Disables the advertisement of speed, duplex mode, and flow control on a Gigabit Ethernet interface.
Step 4	**channel-group** *channel-group-number* **Example:** `Router(config-if-srv)# channel-group 1`	Assigns a port-channel interface to a port-channel group.

	Command or Action	Purpose
Step 5	**end** **Example:** `Router(config-if-srv)# end`	Exits the command-line interface and enters the privileged EXEC mode.

Load Balancing the EVCs on Port Channels

The load balancing of the EVCs in a port channel aims to load balance traffic across the member links of that port channel when the EVCs are configured. In the VLAN-based load-balancing method, if you do not assign EVCs to a member link, they will be statically mapped to one of the active port-channel member links, which will result in the outgoing traffic being limited to the bandwidth of the member link. In the flow-based load-balancing method, the traffic is distributed across all member links.

EFPs are configured under a port channel. The traffic, which is carried by the EFPs, is load balanced across member links. Ingress traffic for a single EVC can arrive on any member of the bundle. All the egress traffic pertaining to an EFP uses only one of the member links. Load balancing is achieved by grouping EFPs and assigning them to a member link. The Cisco ASR 1000 Layer 2 port-channel interface supports flow-based load balancing by default. In default load balancing, you have no control over how the EFPs are grouped together, and sometimes, the EFP grouping may not be ideal. To avoid this, use VLAN-based manual load balancing to control the EFP grouping.

Flow-Based Load Balancing

Flow-based load balancing is the default load-balancing method that is applied on a specific system level or a specific port channel level. However, EVC port channel also supports the VLAN-based load-balancing method. You can configure either of the load balancing methods globally for all the port channels or directly on specific port channels.

Load-Balancing Algorithm

When you select flow-based load balancing, a global flow-based load-balancing algorithm is applied to the port channel by default. Using the packet header information, the corresponding port channel balances the traffic across all the member links. You can configure the device to use one of the following methods to load balance across the port channels:

- Destination IP Address
- Destination MAC Address
- Source exclusive OR (XOR) Destination IP Address
- Source XOR Destination MAC Address
- Source IP Address
- Source MAC Address

The provision that you configure will be applied to all the egress traffic streams on all the port-channel interfaces that have flow-based load balancing.

The following is a sample configuration of the flow-based load-balancing algorithm:

```
Router(config)# port-channel load-balance-hash-algo ?
dst-ip Destination IP
dst-mac Destination MAC
src-dst-ip Source XOR Destination IP Addr
src-dst-mac Source XOR Destination MAC
src-ip Source IP
src-mac Source MAC
```



Note Flow-based load balancing is enabled by default at the global level. You must explicitly configure VLAN-based load balancing; otherwise, the default load-balancing method is flow-based.

To enable flow-based load balancing on a port channel, perform the following steps:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface port-channel** *port-channel number*
4. **load-balancing** {**flow** | **vlan**}
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router# enable`	Enables the privileged EXEC mode.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters the global configuration mode.
Step 3	**interface port-channel** *port-channel number* **Example:** `Router(config)# interface port-channel 1`	Specifies the port channel to configure, and enters the configuration mode.
Step 4	**load-balancing** {**flow**	**vlan**} **Example:** `Router(config-if)# load-balancing flow`	Applies the load-balancing method to the specific port channel.

	Command or Action	Purpose
Step 5	**end** **Example:** `Router(config-if)# end`	Exits the configuration mode.

Configuring Flow Based Load Balancing

To configure flow-based load balancing on an EVC port channel, perform the following steps.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface port-channel** *channel-number*
4. **port-channel load-balance flow-based**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router# enable`	Enables the privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters the global configuration mode.
Step 3	**interface port-channel** *channel-number* **Example:** `Router(config)# interface port-channel 1`	Creates the port-channel interface.
Step 4	**port-channel load-balance flow-based** **Example:** `Router(config-if)# port-channel load-balance flow-based`	Configures the specified port-channel interface in the flow-based load-balancing mode.
Step 5	**end**	Exits the configuration mode.

VLAN-Based Manual Load Balancing

Configuring VLAN Based Manual Load Balancing

Perform this task to link a VLAN port channel, and to enable VLAN load balancing on port channels:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface port-channel** *channel-number*
4. **no ip address**
5. **no negotiation auto**
6. **port-channel load-balancing link 1**
7. **backup link 2**
8. **service-instance 100,300**
9. **port-channel load-balancing link 2**
10. **backup link 1**
11. **service-instance 200,400**
12. **load-balancing vlan**
13. **service instance** *id* **ethernet**
14. **encapsulation** { **default** | **untagged** | **dot1q** *vlan-id* [**second-dot1q** *vlan-id*] }
15. **service instance** *id* **ethernet**
16. **encapsulation** { **default** | **untagged** | **dot1q** *vlan-id* [**second-dot1q** *vlan-id*] }
17. **service instance** *id* **ethernet**
18. **encapsulation** { **default** | **untagged** | **dot1q** *vlan-id* [**second-dot1q** *vlan-id*] }
19. **service instance** *id* **ethernet**
20. **encapsulation** { **default** | **untagged** | **dot1q** *vlan-id* [**second-dot1q** *vlan-id*] }
21. **interface** *type number*
22. **no ip address**
23. **no negotiation auto**
24. **channel-group** *channel-number* **link** *link-number*
25. **interface** *type number*
26. **no ip address**
27. **no negotiation auto**
28. **channel-group** *channel-number* **link** *link-number*
29. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables the privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters the global configuration mode.
Step 3	**interface port-channel** *channel-number* **Example:** `Router(config)# interface port-channel 1`	Enters the interface configuration mode and defines the interface as a port channel.
Step 4	**no ip address** **Example:** `Router(config-if)# no ip address`	Removes an IP address or disables IP processing.
Step 5	**no negotiation auto** **Example:** `Router(config-if)# no negotiation auto`	Disables the advertisement of speed, duplex mode, and flow control on a port-channel interface.
Step 6	**port-channel load-balancing link 1** **Example:** `Router(config-if)# port-channel load-balancing link 1`	Enables port-channel load balancing on link 1.
Step 7	**backup link 2** **Example:** `Router(config-if)# backup link 2`	Assigns a backup link 2.
Step 8	**service-instance 100,300** **Example:** `Router(config-if)# service-instance 100,300`	Creates a service instance on the interface.
Step 9	**port-channel load-balancing link 2** **Example:** `Router(config-if)# port-channel load-balancing link 2`	Enables port-channel load balancing on link 2.
Step 10	**backup link 1** **Example:** `Router(config-if)# backup link 1`	Assigns a backup link 1.

	Command or Action	Purpose
Step 11	**service-instance 200,400** **Example:** `Router(config-if)# service-instance 200,400`	Creates a service instance on the interface.
Step 12	**load-balancing vlan** **Example:** `Router(config-if)# load-balancing vlan`	Enables port-channel load balancing on the router.
Step 13	**service instance** *id* **ethernet** **Example:** `Router(config-if)# service instance 100 ethernet`	Creates a service instance on the interface.
Step 14	**encapsulation** { **default** | **untagged** | **dot1q** *vlan-id* [**second-dot1q** *vlan-id*] } **Example:** `Router(config-if-srv)# encapsulation dot1q 100`	Defines the matching criteria to be used in order to map the ingress dot1q frames on an interface to the appropriate service instance.
Step 15	**service instance** *id* **ethernet** **Example:** `Router(config-if)# service instance 200 ethernet`	Creates a service instance on the interface.
Step 16	**encapsulation** { **default** | **untagged** | **dot1q** *vlan-id* [**second-dot1q** *vlan-id*] } **Example:** `Router(config-if-srv)# encapsulation dot1q 200`	Defines the matching criteria to be used in order to map the ingress dot1q frames on an interface to the appropriate service instance.
Step 17	**service instance** *id* **ethernet** **Example:** `Router(config-if)# service instance 300 ethernet`	Creates a service instance on the interface.
Step 18	**encapsulation** { **default** | **untagged** | **dot1q** *vlan-id* [**second-dot1q** *vlan-id*] } **Example:** `Router(config-if-srv)# encapsulation dot1q 300`	Defines the matching criteria to be used in order to map the ingress dot1q frames on an interface to the appropriate service instance.
Step 19	**service instance** *id* **ethernet** **Example:** `Router(config-if)# service instance 400 ethernet`	Creates a service instance on the interface.
Step 20	**encapsulation** { **default** | **untagged** | **dot1q** *vlan-id* [**second-dot1q** *vlan-id*] } **Example:** `Router(config-if-srv)# encapsulation dot1q 400`	Defines the matching criteria to be used in order to map the ingress dot1q frames on an interface to the appropriate service instance.

	Command or Action	Purpose
Step 21	**interface** *type number* **Example:** `Router(config)# interface gigabitethernet0/2/6`	Specifies the interface type and number.
Step 22	**no ip address** **Example:** `Router(config-if)# no ip address`	Removes an IP address or disables IP processing.
Step 23	**no negotiation auto** **Example:** `Router(config-if)# no negotiation auto`	Disables the advertisement of speed, duplex mode, and flow control on a port-channel interface.
Step 24	**channel-group** *channel-number* **link** *link-number* **Example:** `Router(config-if)# channel-group 1 link 1`	Assigns a Gigabit Ethernet interface to the specified channel group and link.
Step 25	**interface** *type number* **Example:** `Router(config)# interface gigabitethernet0/2/7`	Specifies the interface type and number.
Step 26	**no ip address** **Example:** `Router(config-if)# no ip address`	Removes an IP address or disables IP processing.
Step 27	**no negotiation auto** **Example:** `Router(config-if)# no negotiation auto`	Disables the advertisement of speed, duplex mode, and flow control on a port-channel interface.
Step 28	**channel-group** *channel-number* **link** *link-number* **Example:** `Router(config-if)# channel-group 1 link 1`	Assigns a Gigabit Ethernet interface to the specified channel group and link.
Step 29	**end** **Example:** `Router(config-if)# end`	Exits the interface configuration mode, and returns to the privileged EXEC mode.

Example: Ethernet Virtual Connections on Port Channels

Example: Ethernet Virtual Connections on Port Channels

The following example shows how to configure flow-based load balancing on a port-channel interface:

```
Router# enable
Router# configure terminal
Router(config)# interface Port-channel 1
Router(config-if)# port-channel load-balance flow-based
Router(config-if)# end
```

The following example shows how to configure QoS on an EVC port channel:

```
Router> enable
Router# configure terminal
Router(config)# class-map c1-cos2-cos4
match cos 2 match vlan 1
Router(config)# policy-map p1-ingress
Router(config-pmap)# class c1-cos2-cos4
Router(config-pmap-c)# police cir 100000 conform-action set-cos-transmit 4
Router(config)# interface Port-channel1
Router(config-if)# no ip address
Router(config-if)# no negotiation auto
Router(config-f)# load-balancing vlan
Router(config-if)# service instance 1 ethernet
Router(config-if-srv)# encapsulation dot1q 1
Router(config-if-srv)# service-policy input p1-ingress
Router(config-if)# port-channel load-balance link 1
Router(config-if)# backup link 2
Router(config-if)# service-instance 1
Router(config-if)# interface GigabitEthernet3/0/3
Router(config-if)# no ip address
Router(config-if)# no shutdown
Router(config-if)# negotiation auto
Router(config-if)# channel-group 1 link 1
Router(config-if)# interface GigabitEthernet5/0/0
Router(config-if)# no ip address
Router(config-if)# no shutdown
Router(config-if)# negotiation auto
Router(config-if)# channel-group 1 link 2
Router(config-if)# bridge-domain 1
Router(config-if)# member Port-channel1 service-instance 1
```

The following example shows how to use the **show running-config interface port-channel** *channel-number* command to verify an EVC port channel's per-flow load-balancing configuration:

```
Router# enable
Router# configure terminal
Router(config)# interface Port-channel 2
Router(config-if)# port-channel load-balance flow-based
Router(config-if)# end
Router# show running-config interface Port-channel 2

Building configuration...
Current configuration : 113 bytes
!
interface Port-channel2
band width 1000000
no ip address
 port-channel load-balance flow-based
end
```

Configuring LACP

To configure the channel mode for each individual link in the LACP port channel, perform the following steps.

SUMMARY STEPS

1. **config t**
2. **interface** *type slot/port*
3. **channel-group number mode** {**active** | **on** | **passive**}
4. **lacp** {**port-priority** | **rate**}
5. **lacp** {**ratefast** | **normal**}

DETAILED STEPS

	Command or Action	Purpose

Step 1	**config t**
	
	Example:
	`Router# config t`
	`Router(config)#`
		Enters the configuration mode.

Step 2	**interface** *type slot/port* **Example:** `Router(config)# interface GigabitEthernet0/2/0`	Specifies the interface to configure, and enters the interface configuration mode.
Step 3	**channel-group number mode** {**active**	**on**	**passive**} **Example:** `Router(config-if)# channel-group 1 mode active`	Specifies the port mode for the link in a port channel. After the LACP is enabled, configure each link or the entire channel as active or passive. When you run port channels with no associated aggregation protocol, the port channel mode is always on.
Step 4	**lacp** {**port-priority**	**rate**} **Example:** `Router(config-if)# lacp port-priority 4000`	Configures the port priority for use with the LACP.
Step 5	**lacp** {**ratefast**	**normal**} **Example:** `Router(config-if)# lacp rate fast`	Configures the fast rate (one second) at which the LACP control packets are sent to an LACP-supported interface. To reset the timeout rate to its default, use the **no** form of this command.

Configuring the LACP Mode per Port Channel Interface

To configure the LACP mode on an individual port-channel interface, perform the following steps.

SUMMARY STEPS

1. **interface port-channel** *port-channel number*
2. **lacp** {**fast switch-over** | **max-bundle** | **min-bundle**}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**interface port-channel** *port-channel number* **Example:** `Router(config)# interface port-channel 1`	Specifies the port channel to configure, and enters the configuration mode.
Step 2	**lacp** {**fast switch-over** | **max-bundle** | **min-bundle**} **Example:** `Router (config-int)# lacp max-bundle 4`	**max-bundle**-Specifies the maximum number of ports to bundle in this port channel. **fast switch-over**-Enables fast switchover on this port channel. **min-bundle**-Specifies the minimum number of ports to bundle in this port channel. Configures four active links on the port channel. The remaining links are in standby mode. Traffic is load-balanced among the active links.