Configuring TAP Aggregation and MPLS Stripping

This chapter describes how to configure TAP aggregation and MPLS stripping on Cisco NX-OS devices.

This chapter contains the following sections:

About TAP Aggregation

Network TAPs

You can use various methods to monitor packets. One method uses physical hardware test access points (TAPs).

Network TAPs can be extremely useful in monitoring traffic because they provide direct inline access to data that flows through the network. In many cases, a third party monitors the traffic between two points in the network. If the network between points A and B consists of a physical cable, a network TAP might be the best way to accomplish this monitoring. The network TAP has at least three ports: an A port, a B port, and a monitor port. A TAP inserted between the A and B ports passes all traffic through unimpeded, but it also copies that same data to its monitor port, which could enable a third party to listen.

TAPs have the following benefits:

- They can handle full-duplex data transmission.
- They are unobtrusive and not detectable by the network (with no physical or logical addressing).
- Some TAPs support full inline power with the capability to build a distributed TAP.

If you are trying to gain visibility into the server-to-server data communication at the edge or virtual edge of your network or to provide a copy of traffic to the Intrusion Prevention System (IPS) appliance at the Internet edge of your network, you can use network TAPs nearly anywhere in the environment. However, this deployment can add significant costs, operation complexities, and cabling challenges in a large-scale environment.

TAP Aggregation

TAP aggregation is an alternative solution to help with monitoring and troubleshooting tasks in the data center. It works by designating a device to allow the aggregation of multiple test access points (TAPs) and to connect to multiple monitoring systems. TAP aggregation switches link all of the monitoring devices to specific points in the network fabric that handle the packets that need to be observed.

In the TAP aggregation switch solution, a Cisco Nexus 9000 Series switch is connected to various points in the network at which packet monitoring is advantageous. From each network element, you can use switched port analyzer (SPAN) ports or optical TAPs to send traffic flows directly to this TAP aggregation switch. The TAP aggregation switch is directly connected to all of the analysis tools used to monitor the events in the network fabric. These monitoring devices include remote monitor (RMON) probes, application firewalls, IPS devices, and packet sniffer tools.

You can configure the TAP aggregation switch to filter specific traffic and redirect it to one or more tools. In order to redirect the traffic to multiple interfaces, a multicast group is created internally on the switch, and the interfaces that are part of the redirect list are added as member ports. When an access control list (ACL) policy with the redirect action is applied to an interface, the traffic matching the ACL rule is redirected to the internal multicast group that is created.

Guidelines and Limitations for TAP Aggregation

Note For scale information, see the release-specific *Cisco Nexus 9000 Series NX-OS Verified Scalability Guide*.

TAP aggregation has the following guidelines and limitations:

- TAP aggregation:
  - Supported on all Cisco Nexus 9000 Series switches and the 3164Q, 31128PQ, 3232C, and 3264Q switches.
  - Supported on 100G ports.
  - Supports only on switch ports and only in the ingress direction.
  - Supports IPv4 ACLs with UDF-based match for Cisco Nexus 9200, 9300, and 9300-EX Series switches.
  - Supported on Cisco Nexus 9300-FX, 9300-FX2, 9300-FX3, 9300-GX, 9300-GX2, 9500-EX, and 9500-FX platform switches.
  - Maximum redirect ports supported are 32 interfaces.
- Beginning with Cisco NX-OS Release 9.2(1), TAP aggregation filters on MPLS tags are supported on the following Cisco Nexus platform switches:

- Cisco Nexus 9000 platform switches, including the 9700-EX and 9700-FX line cards.
- Cisco Nexus 9200 platform switches.
- Cisco Nexus 9300 platform switches.
- Cisco Nexus 9500 switches.

- TAP aggregation filters on MPLS tags are not supported on the following Cisco Nexus Series switches, line cards, and fabric modules:

Table 1: Cisco Nexus 9000 Series Switches

Cisco Nexus 3164Q-40GE	Cisco Nexus 9372PX	Cisco Nexus 9372PX-E
Cisco Nexus 9372TX	Cisco Nexus 9372TX-E	Cisco Nexus 9332PQ
Cisco Nexus 3232C	Cisco Nexus 93120TX	Cisco Nexus 31128PQ
Cisco Nexus 3264Q-S	—	—

Table 2: Cisco Nexus 9000 Series Line Cards and Fabric Modules

N9K-M6PQ	N9K-X9632PC-QSFP100	N9K-X9536PQ
N9K-X9432C-S	N9K-C93128TX	N9K-C9396PX
N9K-X9432PQ	N9K-X9464TX	—

- Cisco Nexus 9700-EX and 9700-FX line cards support TAP aggregation with IPv4, IPv6, and MAC ACLs.
- Only Layer 2 interfaces support the TAP aggregation policy. You can apply the policy to a Layer 3 interface, but the policy becomes nonfunctional.
- The redirect port must be part of the same VLAN as the source (TAP) port.
- Each rule must be associated with only one unique match criterion.
- When you enter a list of interfaces for the TAP aggregation policy, you must separate them with commas but no spaces. For example, port-channel50, ethernet1/12, port-channel20.
- When you specify target interfaces in a policy, make sure that you enter the whole interface type and not an abbreviated version. For example, make sure that you enter **ethernet1/1** instead of **eth1/1** and **port-channel50** instead of **po50**.
- HTTP requests with *tcp-option-length* and *VLAN ID* filters simultaneously are not supported. Traffic match against ACE may not work if you configure both filters at a time.
- Beginning with Cisco NX-OS Release 10.2(1)F, the TAP aggregation feature is licensed and requires you to configure feature tap-aggregation before configuring related CLIs. However, this feature is auto-generated during ISSU infra-convert phase of sysmgr if any tap-aggregation dependent CLI usage is found in the earlier configurations. This feature is supported on all Cisco Nexus 9000 Series switches. For more information about licensing, refer to *Cisco Nexus 9000 NX-OS Smart Licensing Using Policy Guide*.

- Beginning with Cisco NX-OS Release 10.2(2)F, ensure that you configure the **mode tap-aggregation** command before attaching TapAgg ACLs on L2 interface.
- When configuring ACL entries with redirect to port-channels that are yet to be configured, the user must take care to configure the specified port-channels at a later point of time.
- Beginning with Cisco NX-OS Release 10.3(1)F, on the interface with selective Q-in-Q trunk mode the provider VLAN tagging is supported on the Cisco Nexus 9300-GX, N9K-C9504-FM-G, and N9K-C9508-FM-G switches and N9K-X9716D-GX line cards with below limitations:
  - If VXLAN is enabled, this feature is not supported.
  - You can have maximum 7000 outer VXLAN late entries at overall system level and 4000 entries per port.
- To allow double VLAN tags on ingress interface, the **switchport trunk allow-multi-tag** command must be configured correctly as mentioned below:
  - On Cisco Nexus 9300-FX2 switches, this command must be used only if NDB is configured.
  - On Cisco Nexus 9300-GX/GX2 switches, this command is not required if NDB is configured.

About Deduplication

From Cisco NX-OS Release 10.4(1)F, deduplication feature can be used to remove duplicate copies of the data flow when they are going through switches in Nexus Data Broker (NDB).

Due to the continuous rise in data flow, navigating in the networks and the data processed in different applications leads to data duplication both in source and target. For efficient data management, security, and storage you can use deduplication to remove duplicate data.

The deduplication feature eliminates the duplicate traffic which is received from traffic analysers or data storage tools. It identifies duplicate flows which are transferred in the NDB switch. Deduplication supports two model types:

- Inline deduplication Model (Interface Mode)

  You can configure the inline model on existing NDB switches. This eliminates duplicate packets using interfaces. Initial interface packets are forwarded and the duplicate interface packets from other interfaces are dropped.

- Deduplication on a Stick Model (VLAN Mode)

  In this model, the packet flow is bound to a specific VLAN. The packet flow is permited on the first VLAN and duplicate flows on other VLANs are restricted. The packets are tagged with specific VLAN when the flow enters NDB switches. Each packet is tagged with a unique VLAN, the packets with VLAN tag are forwarded to the deduplication switch, and duplicate flow is removed on that switch.

Guidelines and Limitations

The below mentioned are the guidelines and limitations for the deduplication:

- You can configure deduplication for inner packet flows only.

- TCP and UDP flows are supported for flow deduplication.
- Deduplication is supported for local SPAN and Optical TAP flows sessions.
- Deduplication can be performed for VXLAN and GRE tunnel packets with a single encap.
- Deduplication is supported for Cisco Nexus 9300-FX2/FX3 and 9300-GX/GX2 platorm switches.
- Deduplication is not supported for:
  - Cisco Nexus 9500 platorm switches
  - Cisco Nexus 9300-C, 9300-EX, 9300-FX and EOR switches
  - Cisco N9K-C9408 switch
- You cannot configure deduplication for duplicate packets on the same interface such as ERSPAN. It may terminate more than a single ERSPAN session on an interface or SVI. This impacts the copy of same flow ends up on the same interface or VLAN. A switch cannot differentiate different ERSPAN sessions for deduplication.
- You cannot delete duplication flow for short lived flows.
- Flow packets from different VRFs with similar 5 tuples cannot be deleted. As deduplication is for the flows performed using 5-tuple of a packet. It cannot identify VRF packets.
- Deduplication stick model (VLAN model) is not supported for tunnel termination, and Q-in-Q ports.
- Deduplication is not supported for IPv6 and multicast flows.
- Dynamic aging is supported only for 128k flows.
- During ISSU, deduplication is disabled by default. Post ISSU all flows are cleared and refreshed.
- Deduplication to function appropriately on tunnel traffic, ensure that you enable flow terminate.

Configuring Deduplication

From Cisco NX-OS Release 10.4(1)F, deduplication feature can be used to remove duplicate copies of the data flow when they are going through switches in Nexus Data Broker (NDB).

Ensure that you reload switch after configuring deduplication, for the configuration changes to be effective.

SUMMARY STEPS

1. **configure terminal**
2. **tap-aggregation flow-deduplication**
3. (Optional) **absolute-timer***time in minutes*
4. (Optional) **dynamic-timer***time in milli seconds*
5. **mode** {**vlan**|**interface**}
6. **clear hardware deduplication statistics**{**slot**|**module in number**}

DETAILED STEPS

	Command or Action	Purpose

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

Step 2	**tap-aggregation flow-deduplication** **Example:** `switch(config-dedup)#`	Enable flow-deduplication feature on a switch.

Step 3	(Optional) **absolute-timer***time in minutes*
	Example:
	`switch(config)# absolute-timer 10`
	`switch(config-dedup)#`
		Allows to configure absolute timer to deduplication of packet flows. The interval range is from 0 to 1440 minutes.

Step 4	(Optional) **dynamic-timer***time in milli seconds*
	Example:
	`switch(config)# dynamic-timer 2000`
	`switch(config-dedup) mode interface#`
		Allows to configure dynamic timer to deduplication of packet flows. The interval range is from 0 to 300000 milli seconds.

Step 5	**mode** {**vlan**|**interface**}
	Example:
	`switch(config)# interface`
	`switch(config-dedup)#`
		Allows to configure deduplication on required mode.
		Note Ensure to save configuration and reload the switch to configure deleting duplication on the switch.

Step 6	**clear hardware deduplication statistics**{**slot**|**module in number**}	To clear deduplication on the required slot or module. The slot or module number range is from 1 to 30.

Example

Below shows the sample output for reference:

```
switch# show hardware deduplication summary
slot 1
=======
Deduplication      : Enabled
Dedup Mode         : Interface
Dynamic timer                : 200000 milliseconds
Absolute timer               : 5 minutes
Max Supported Flows          : 240K
Total number of learned flows : 240000
Total number of dropped bytes : 65698869600
```

```
switch# show hardware deduplication detail
slot1
================================================================================
   Dedup Flows
================================================================================
SourceIP  Destination IP  Ports(Src:Dst) Protocol    Interface     Learn-time
================================================================================
```

```
33.1.1.2  12.1.1.2   3000 :3001  6  Eth1/1  07/28/2023 11:47:09.532376
55.1.1.2  12.1.1.2   15000:15001  17     Eth1/1  07/28/2023 11:47:09.532229
11.1.1.2  12.1.1.2   1841 :1842  6  Eth1/1  07/28/2023 11:47:09.532340
1.22.1.2  1.12.1.2   2000 :2001  6  Eth1/1  07/28/2023 11:47:09.532428
1.44.1.2  1.12.1.2   4000 :4001  6  Eth1/23  07/28/2023 11:47:09.532133

switch#show hardware deduplication age-history
slot 1
================================================================================
    Dedup Flows
================================================================================
Source  Destination Ports  Protocol Interface Timer    Learn-Time    Aged-Time
IP  IP   (Src:Dst)
================================================================================
1.44.1.2 1.12.1.2   4000:4001 6  Eth1/17 Dynamic   08/05/2023 2:24:49.26020      08/05/2023
 12:33:29.21904
33.1.1.2  12.1.1.2  3000:3001       6  Eth1/27 Dynamic  08/05/2023 12:24:49.126246
08/05/2023 12:33:29.21945
55.1.1.2  12.1.1.2  15000:15001    17  Eth1/5 Dynamic    08/05/2023 12:24:49.26070
08/05/2023 12:33:29.21957
1.22.1.2  1.12.1.2   2000 :2001     6  Eth1/5 Dynamic    08/05/2023 12:24:49.26115
08/05/2023 12:33:29.21969
11.1.1.2  12.1.1.2   1841 :1842     6  Eth1/17 Dynamic   08/05/2023 12:24:49.25949
08/05/2023 12:33:29.21979
```

About MPLS Stripping

The ingress ports of Cisco Nexus 9000 Series switches receive various Multiprotocol Label Switching (MPLS) packet types. Each data packet in an MPLS network has one or more label headers. These packets are redirected on the basis of a redirect access control list (ACL).

A label is a short, four-byte, fixed-length, locally significant identifier that is used to identify a Forwarding Equivalence Class (FEC). The label that is put on a particular packet represents the FEC to which that packet is assigned. It has the following components:

- Label—Label value (unstructured), 20 bits
- Exp—Experimental use, 3 bits; currently used as a class of service (CoS) field
- S—Bottom of stack, 1 bit
- TTL—Time to live, 8 bits

Standard network monitoring devices cannot monitor and analyze the MPLS traffic. You need to enable the MPLS strip feature to allow the standard network monitoring tools to monitor the MPLS traffic. This feature strips off the MPLS label headers of the traffic and redirects the traffic to the monitoring devices.

Guidelines and Limitations for MPLS Stripping

Note For scale information, see the release-specific *Cisco Nexus 9000 Series NX-OS Verified Scalability Guide*.

MPLS stripping has the following guidelines and limitations:

- Cisco Nexus 9700-EX and 9700-FX line cards do not support MPLS stripping.

- Beginning from Cisco NX-OS Release 10.2(1)F, **feature tap-aggregation** must be enabled for all Tap Aggregation and stripping functions.
- Disable all Layer 3 and vPC features before you enable MPLS stripping.
- Static MPLS, MPLS segment routing, and MPLS stripping cannot be enabled at the same time.
- Only the ingress interfaces involved in MPLS stripping must have TAP aggregation enabled.
- You must configure the TAP aggregation ACL with a redirect action on the ingress interface to forward the packet to the desired destination.
- Post MPLS strip, SMAC changes to switch mac (**show vdc**) and DMAC is set to **00:00:00:ab:cd:ef**.
- The egress interface where stripped packets will exit must be an interface that has VLAN 1 as an allowed VLAN. We recommend that you configure the egress interface as a trunk with all VLANs allowed by default.
- Stripping is based on IP PACL, and you cannot use MAC-ACL for stripping.
- MPLS stripping is supported only for IPv4 traffic.
- Port-channel load balancing is supported for MPLS stripped packets.
- Layer 3 header-based hashing and Layer 4 header-based hashing are supported, but Layer 2 header-based hashing is not supported.
- During MPLS stripping, the incoming VLAN is not preserved.
- Cisco Nexus 9200, 9300-EX, and 9300-FX platform switches support tagging of VLANs to packets going out of redirect ports. The ingress/egress ports can either be ethernet or port channel. The VLAN tag is derived from the incoming port configuration. The new ACL on the ingress interface should not be associated with a VLAN value different from the interface VLAN value.
- For every ACE (under an ACL associated with a particular VLAN) with a unique redirect port list, we allocate a hardware entry. The current hardware limit for the number of ACEs is 50 and you cannot configure more than 50 such ACEs.
- MPLS strip is only supported for Layer 3 packets under the MPLS label stack.
- Beginning with Cisco NX-OS Release 10.2(2)F, EoMPLS label stripping is supported only on Cisco Nexus 9300-EX platform switches. However, VPLS strip and control-word packet strip is not supported.
- Beginning with Cisco NX-OS Release 10.2(3)F, OFM-based MPLS stripping is added. The new OFM-based MPLS stripping and legacy implementation cannot co-exist. For more information, see the OFM-based MPLS header strip section under Configuring Header Stripping Features for Nexus Data Broker.
- Use the new OFM-based MPLS stripping feature only if the deployment needs co-existence of MPLS stripping with any other type of header stripping such as VXLAN, iVXLAN, GRE, and ERSPAN headers.

  The existing MPLS stripping feature will continue to support MPLS stripping when co-existence is not needed with other stripping features.
- Beginning with Cisco NX-OS Release 10.3(2)F, EoMPLS label stripping is also supported on Cisco Nexus 9300-FX ToR switches.

Configuring TAP Aggregation

Enabling TAP Aggregation for Line Cards

Beginning with Cisco NX-OS Release 7.0(3)I7(2), you can enable TAP aggregation for Cisco Nexus 9500 platform switches with 9700-EX and 9700-FX line cards.

SUMMARY STEPS

1. **configure terminal**
2. [**no**] **hardware acl tap-agg**
3. (Optional) **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose

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

Step 2	[**no**] **hardware acl tap-agg** **Example:** `switch(config)# hardware acl tap-agg`	Enables TAP aggregation for Cisco Nexus 9700-EX and 9700-FX line cards. This command is also needed on Cisco Nexus 9300-GX and 9300-GX2 platform switches and may require reload.
Step 3	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Configuring a TAP Aggregation Policy

You can configure a TAP aggregation policy on an IP access control list (ACL) or on a MAC ACL.

Before you begin

You must configure the ACL TCAM region size for IPv4 port ACLs or MAC port ACLs using the **hardware access-list tcam region** {*ifacl* | *mac-ifacl*} command. Configure the ACL TCAM region size for IPv6 port ACLs using the command, **hardware access-list team region ipv6-ifcal**.

For information, see the "Configuring ACL TCAM Region Sizes" in the Cisco Nexus 9000 Series NX-OS Security Configuration Guide.

Note By default the region size for both ifacl and mac-ifacl is zero. You need to allocate enough entries to the ifacl or mac-ifacl region to support TAP aggregation.

SUMMARY STEPS

1. **configure terminal**
2. **feature tap-aggregation**
3. Enter one of the following commands:
   - **ip access-list** *access-list-name*
   - **mac access-list** *access-list-name*
4. (Optional) **statistics per-entry**
5. [**no**] **permit** *protocol source destination* **redirect** *interfaces*
6. (Optional) Enter one of the following commands:
   - **show ip access-lists** [*access-list-name*]
   - **show mac access-lists** [*access-list-name*]
7. (Optional) **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose

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

Step 2	**feature tap-aggregation**
	Example:
	`switch(config)# feature tap-aggregation`
	`switch(config)#`
		Allows you to configure to CLIs related to tap-aggregation.
		Note Beginning with Cisco NX-OS Release 10.2(1)F, for software upgrades from earlier releases to the newer NX-OS release with this feature, if ISSU is completed on a supported matrix, the feature tap-aggregation configuration is automatically generated.

Step 3	Enter one of the following commands:
	• **ip access-list** *access-list-name*
	• **mac access-list** *access-list-name*
	Example:
	`switch(config)# ip access-list test`
	`switch(config-acl)#`
	
	`switch(config)# mac access-list mactap1`
	`switch(config-mac-acl)#`
		Creates an IPACL and enters IP access list configuration mode or creates a MAC ACL and enters MAC access list configuration mode.

	Command or Action	Purpose
Step 4	(Optional) **statistics per-entry** **Example:** `switch(config-acl)# statistics per-entry`	Starts recording statistics for how many packets are permitted or denied by each entry.
Step 5	[**no**] **permit** *protocol source destination* **redirect** *interfaces* **Example:** `switch(config-acl)# permit ip any any redirect ethernet1/8`	Creates an IP or MAC ACL rule that permits traffic to be redirected per its conditions. The **no** version of this command removes the permit rule from the policy. **Note** When you enter an interface for the TAP aggregation policy, do not abbreviate it. When you enter a list of interfaces, separate them with commas but no spaces.

Step 6	(Optional) Enter one of the following commands:
	• **show ip access-lists** [*access-list-name*]
	• **show mac access-lists** [*access-list-name*]
	Example:
	`switch(config-acl)# show ip access-lists test`
	`switch(config-mac-acl)# show mac access-lists mactap1`
		Displays all IPv4 or MAC ACLs or a specific IPv4 or MAC ACL.

Step 7	(Optional) **copy running-config startup-config** **Example:** `switch(config-acl)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Attaching a TAP Aggregation Policy to an Interface

You can apply an ACL configured with TAP aggregation to a Layer 2 interface.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *type slot/port*
3. **switchport**
4. Enter one of the following commands:
   - [**no**] **ip port access-group** *access-list-name* **in**
   - [**no**] **mac port access-group** *access-list-name* **in**
5. (Optional) **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose

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

Step 2	**interface** *type slot/port*
	Example:
	`switch(config)# interface ethernet 2/2`
	`switch(config-if)#`
		Enters interface configuration mode for the specified interface.

Step 3	**switchport** **Example:** `switch(config-if)# switchport`	Changes a Layer 3 interface to a Layer 2 interface. **Note** Make sure that the interface is a Layer 2 interface.

Step 4	Enter one of the following commands:
	• [**no**] **ip port access-group** *access-list-name* **in**
	• [**no**] **mac port access-group** *access-list-name* **in**
	Example:
	`switch(config-if)# ip port access-group test in`
	`switch(config-if)# mac port access-group test in`
		Applies an IPv4 or MAC ACL configured with TAP aggregation to the interface. The **no** form of this command removes the ACL from the interface.

Step 5	(Optional) **copy running-config startup-config** **Example:** `switch(config-if)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Configuring Selective Q-in-Q with Provider VLANs

Before you begin

You must configure provider VLANs

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport**
4. **switchport mode trunk**
5. Enter one of the following commands:
   - **switchport vlan mapping** *vlan-id-range***dot1q-tunnel** *outer vlan-id*
   - **switchport vlan mapping all dot1q-tunnel** *outer vlan-id*
6. **switchport trunk allowed vlan** *vlan_list*

7. Enter one of the following commands:
   - [**no**] **ip port access-group** *access-list-name* **in**
   - [**no**] **mac port access-group** *access-list-name* **in**
8. (Optional) **mode tap-aggregation**
9. (Optional) **copy running-config startup-config**
10. switch(config-if)# **exit**
11. (Optional) switch(config-if)# **show interfaces** *interface-id* **vlan mapping**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `switch(config)# interface Ethernet1/1`	Enters interface configuration mode for the interface connected to the service provider network. You can enter a physical interface or an EtherChannel port channel.
Step 3	**switchport** **Example:** `switch(config if)# switchport`	Sets the interface as a Layer 2 switching port.
Step 4	**switchport mode trunk** **Example:** `switch(config-if)# switchport mode trunk`	Sets the interface as a Layer 2 trunk port.
Step 5	Enter one of the following commands: • **switchport vlan mapping** *vlan-id-range***dot1q-tunnel** *outer vlan-id* • **switchport vlan mapping all dot1q-tunnel** *outer vlan-id* **Example:** `switch(config-if)# switchport vlan mapping all dot1q-tunnel 300`	Enter the VLAN IDs to be mapped: • vlan-id-range—The customer VLAN ID range(C-VLAN) entering the switch from the customer network. The range is from 1 to 4094. You can enter a string of VLAN-IDs. • outer vlan-id—Enter the outer VLAN ID (S-VLAN) of the service provider network. The range is from 1 to 4094.
Step 6	**switchport trunk allowed vlan** *vlan_list* **Example:** `switch(config-if)# switchport trunk allowed vlan 300`	Sets the allowed VLANs for the trunk interface.
Step 7	Enter one of the following commands: • [**no**] **ip port access-group** *access-list-name* **in** • [**no**] **mac port access-group** *access-list-name* **in**	Applies an IPv4 or MAC ACL configured with TAP aggregation to the interface. The **no** form of this command removes the ACL from the interface.

	Command or Action	Purpose

	Example:
	`switch(config-if)# ip port access-group test in`
	`switch(config-if)# mac port access-group test in`	

Step 8	(Optional) **mode tap-aggregation**
	Example:
	`switch(config-if)# mode tap-aggregation`
	`switch(config-if)# no shutdown`
		Allows the attachment of an ACL with the tap aggregation policy to the interface.

Step 9	(Optional) **copy running-config startup-config** **Example:** `switch(config-if)# copy running-config startup-config`	Copies the running configuration to the startup configuration.
Step 10	switch(config-if)# **exit**	Exits the configuration mode.
Step 11	(Optional) switch(config-if)# **show interfaces** *interface-id* **vlan mapping**	Verifies the mapping configuration.

Verifying the TAP Aggregation Configuration

To display the TAP aggregation configuration information, perform one of the following tasks.

Command	Purpose
show ip access-lists [*access-list-name*]	Displays all IPv4 ACLs or a specific IPv4 ACL.
show mac access-lists [*access-list-name*]	Displays all MAC ACLs or a specific MAC ACL.

Configuration Example for TAP Aggregation

This example shows how to configure a TAP aggregation policy on an IPv4 ACL:

```
switch# configure terminal
switch(config)# feature tap-aggregation
switch(config)# ip access-list test
switch(config-acl)# 10 deny ip 100.1.1/24 any
switch(config-acl)# 20 permit tcp any eq www any redirect port-channel4
switch(config-acl)# 30 permit ip any any redirect
Ethernet1/1,Ethernet1/2,port-channel7,port-channel8,Ethernet1/12,Ethernet1/13
switch(config-acl)# show ip access-lists test
IP access list test
        10 deny ip 100.1.1/24 any
        20 permit tcp any eq www any redirect port-channel4
        30 permit ip any any redirect
Ethernet1/1,Ethernet1/2,port-channel7,port-channel8,Ethernet1/12,Ethernet1/13
```

This example shows how to configure a TAP aggregation policy on a MAC ACL:

```
switch# configure terminal
switch(config)# feature tap-aggregation
switch(config)# mac access-list mactap1
switch(config-mac-acl)# 10 permit any any 0x86dd redirect port-channel1
switch(config-mac-acl)# show mac access-lists mactap1
MAC access list mactap1
        10 permit any any 0x86dd redirect port-channel1
```

This example shows how to attach a TAP aggregation policy to a Layer 2 interface:

```
switch# configure terminal
switch(config)# interface ethernet 1/2
switch(config-if)# ip port access-group test in
switch(config-if)#
```

Configuring MPLS Stripping

Enabling MPLS Stripping

You can enable MPLS stripping globally.

Before you begin

Disable all Layer 3 and vPC features before you enable MPLS stripping.

Attach an ACL with the tap aggregation policy to the Layer 2 interface or port channel using the **mode tap-aggregation** command. For more information, see Attaching a TAP Aggregation Policy to an Interface, on page 11.

SUMMARY STEPS

1. **configure terminal**
2. [**no**] **mpls strip**
3. [**no**] **mpls strip mode dot1q**
4. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose

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

Step 2	[**no**] **mpls strip** **Example:** `switch(config)# mpls strip`	Globally enables MPLS stripping. The **no** form of this command disables MPLS stripping.

	Command or Action	Purpose
Step 3	[**no**] **mpls strip mode dot1q** **Example:** `switch(config)# mpls strip mode dot1q`	Enables VLAN tagging on the packets coming from the redirect port. The VLAN that needs to be tagged must be specified in the ingress port.
Step 4	Required: **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Configuring the Incoming Port for the VLAN Tag

The VLAN tag is derived from the incoming port configuration. The ingress/egress ports can either be ethernet or port channel.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *type slot/port*
3. **switchport**
4. Enter one of the following commands:
   - [**no**] **ip port access-group** *access-list-name* **in**
   - [**no**] **mac port access-group** *access-list-name* **in**
5. Enter one of the following commands:
   - [**no**] **ip port access-group** *access-list-name* **in**
   - [**no**] **mac port access-group** *access-list-name* **in**
6. (Optional) **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose

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

Step 2	**interface** *type slot/port*
	Example:
	`switch(config)# interface ethernet 1/26`
	`switch(config-if)#`
		Enters interface configuration mode for the specified interface.

Step 3	**switchport** **Example:** `switch(config-if)# switchport`	Changes a Layer 3 interface to a Layer 2 interface. **Note** Make sure that the interface is a Layer 2 interface.

	Command or Action	Purpose

Step 4	Enter one of the following commands:
	• [**no**] **ip port access-group** *access-list-name* **in**
	• [**no**] **mac port access-group** *access-list-name* **in**
	Example:
	`switch(config-if)# ip port access-group test in`
	`switch(config-if)# mac port access-group test in`
		Applies an IPv4 or MAC ACL configured with TAP aggregation to the interface. The **no** form of this command removes the ACL from the interface.

Step 5	Enter one of the following commands:
	• [**no**] **ip port access-group** *access-list-name* **in**
	• [**no**] **mac port access-group** *access-list-name* **in**
	Example:
	`switch(config-if)# ip port access-group test in`
	`switch(config-if)# mac port access-group test in`
		Applies an IPv4 or MAC ACL configured with TAP aggregation to the interface. The **no** form of this command removes the ACL from the interface.

Step 6	(Optional) **copy running-config startup-config** **Example:** `switch(config-if)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Adding and Deleting MPLS Labels

The device can learn the labels dynamically whenever a frame is received with an unknown label on a TAP interface. You can also add or delete static MPLS labels.

Before you begin

Configure a TAP aggregation policy and attach the policy to an interface. For more information, see the *Cisco Nexus 9000 Series NX-OS System Management Configuration Guide*.

You must configure the TAP aggregation ACL with a redirect action on the ingress interface to forward the packet to the desired destination.

SUMMARY STEPS

1. **configure terminal**
2. **mpls strip label** *label*
3. (Optional) **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:**	Enters global configuration mode.

	Command or Action	Purpose
	`switch# configure terminal` `switch(config)#`	
Step 2	**mpls strip label** *label* **Example:** `switch(config)# mpls strip label 100`	Adds the specified static MPLS label. The 20-bit value of the label can range from 1 to 1048575. **Note** This CLI is available for all the platform switches specified for the MPLS Stripping feature in the Guidelines and Limitations section, except for the following cloud scale platform switches: • N9K-C93180YC-EX • N9K-C93180YC-FX • N9K-C93240YC-FX2 • N9K-C93180YC-FX3S • N9K-C93600CD-GX The [**no**] **mpls strip label** {*label* | **all**} command deletes the specified static MPLS label. The **all** option deletes all static MPLS labels.
Step 3	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Configuring Destination MAC Addresses

You can configure the destination MAC address for stripped egress frames.

SUMMARY STEPS

1. **configure terminal**
2. **mpls strip dest-mac** *mac-address*
3. (Optional) **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose

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

	Command or Action	Purpose
Step 2	**mpls strip dest-mac** *mac-address* **Example:** `switch(config)# mpls strip dest-mac 1.1.1`	Specifies the destination MAC address for egress frames that are stripped of their headers. The MAC address can be specified in one of the following four formats: • E.E.E • EE-EE-EE-EE-EE-EE • EE:EE:EE:EE:EE:EE • EEEE.EEEE.EEEE
Step 3	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Configuring MPLS Label Aging

You can define the amount of time after which dynamic MPLS labels will age out, if unused.

SUMMARY STEPS

1. **configure terminal**
2. **mpls strip label-age** *age*
3. (Optional) **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose

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

Step 2	**mpls strip label-age** *age* **Example:** `switch(config)# mpls strip label-age 300`	Specifies the amount of time in seconds after which dynamic MPLS labels age out. The range is from 61 to 31622400.
Step 3	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Verifying the MPLS Stripping Configuration

To display the MPLS stripping configuration, perform one of the following tasks:

Command	Purpose
show mpls strip labels [*label* | **all** | **dynamic** | **static**]	Displays information about MPLS labels. You can specify the following options: • *label*—Label to be displayed. • **all**—Specifies that all labels must be displayed. This is the default option. • **dynamic**—Specifies that only dynamic labels must be displayed. • **static**—Specifies that only static labels must be displayed.

This example shows how to display all MPLS labels:

```
switch# show mpls strip labels
MPLS Strip Labels:
    Total      : 3005
    Static     : 5
Legend:     * - Static Label
    Interface - where label was first learned
    Idle-Age  - Seconds since last use
    SW-Counter- Packets received in Software
    HW-Counter- Packets switched in Hardware
-------------------------------------------------------------------------------
     Label    Interface      Idle-Age   SW-Counter       HW-Counter
-------------------------------------------------------------------------------
     4096    Eth1/53/1            15            1              210
     4097    Eth1/53/1            15            1              210
     4098    Eth1/53/1            15            1              210
     4099    Eth1/53/1             7            2              219
     4100    Eth1/53/1             7            2              219
     4101    Eth1/53/1             7            2              219
     4102    Eth1/53/1            39            1              206
     4103    Eth1/53/1            39            1              206
     4104    Eth1/53/1            39            1              206
     4105    Eth1/53/1             1            1              217
     4106    Eth1/53/1             1            1              217
     4107    Eth1/53/1             1            1              217
     4108    Eth1/53/1            15            1              210
*   25000    None <User>          39            1              206
*   20000    None <User>          39            1              206
*   21000    None <User>           1            1              217
```

This example shows how to display only static MPLS labels:

```
switch(config)# show mpls strip labels static
MPLS Strip Labels:
    Total      : 3005
    Static     : 5
Legend:     * - Static Label
    Interface - where label was first learned
```

```
    Idle-Age  - Seconds since last use
    SW-Counter- Packets received in Software
    HW-Counter- Packets switched in Hardware
-------------------------------------------------------------------------------
     Label     Interface      Idle-Age   SW-Counter       HW-Counter
-------------------------------------------------------------------------------
*      300    None <User>         403            0                0
*      100    None <User>         416            0                0
*    25000    None <User>         869            0                0
*    20000    None <User>         869            0                0
*    21000    None <User>         869            0                0
```

Clearing MPLS Stripping Counters and Label Entries

To clear the MPLS stripping counters and label entries, perform these tasks:

Command	Purpose
clear mpls strip label dynamic	Clears dynamic label entries from the MPLS label table.
clear counters mpls strip	Clears all MPLS stripping counters.

The following example shows how to clear all MPLS stripping counters:

```
switch# clear counters mpls strip
switch# show mpls strip labels
MPLS Strip Labels:
    Total      : 15000
    Static     : 2
Legend:     * - Static Label
    Interface - where label was first learned
    Idle-Age  - Seconds since last use
    SW-Counter- Packets received in Software
    HW-Counter- Packets switched in Hardware
-------------------------------------------------------------------------------
     Label     Interface      Idle-Age   SW-Counter       HW-Counter
-------------------------------------------------------------------------------
      4096    Eth1/44             15            0                0
      8192    Eth1/44             17            0                0
     12288    Eth1/44             15            0                0
     16384    Eth1/44             39            0                0
     20480    Eth1/44             47            0                0
     24576    Eth1/44              7            0                0
     28672    Eth1/44              5            0                0
     36864    Eth1/44              7            0                0
     40960    Eth1/44             19            0                0
     45056    Eth1/44              9            0                0
     49152    Eth1/44             45            0                0
     53248    Eth1/44              9            0                0
```

Configuration Examples for MPLS Stripping

This example shows how to add static MPLS labels:

```
switch# configure terminal
switch(config)# mpls strip label 100
```

```
switch(config)# mpls strip label 200
switch(config)# mpls strip label 300
```

Additional References

Related Documents

Related Topic	Document Title
IP ACLs	*Cisco Nexus 9000 Series NX-OS Security Configuration Guide*
MAC ACLs	*Cisco Nexus 9000 Series NX-OS Security Configuration Guide*
Port-channel symmetric hashing	*Cisco Nexus 9000 Series NX-OS Interfaces Configuration Guide*
Remote monitoring (RMON)	Configuring RMON
Switched port analyzer (SPAN)	Configuring SPAN
Troubleshooting	*Cisco Nexus 9000 Series NX-OS Troubleshooting Guide*