

Test Access Points Aggregation and Multiprotocol Label Switch Stripping

- About TAP Aggregation, on page 1
- About MPLS Stripping, on page 7
- Configuring TAP Aggregation, on page 9
- Verifying the TAP Aggregation Configuration, on page 15
- Configuration Example for TAP Aggregation, on page 16
- Configuring MPLS Stripping, on page 16
- Clearing MPLS Stripping Counters and Label Entries, on page 21
- Configuration Examples for MPLS Stripping, on page 21
- Additional References, on page 22

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.

About TAP Aggregation

Test access points (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 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.

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.

Guidelines and Limitations

General Guidelines

- 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	—

- 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.
- 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.
- 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.

Supported Features

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

- 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.
- Cisco Nexus 9700-EX and 9700-FX line cards support TAP aggregation with IPv4, IPv6, and MAC ACLs.

Release Specific Guidelines

- 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.
- 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.
- 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.

Limitations for hardware acl tap-aggCommand

- This command is required on Cisco Nexus 9300-GX platform switches for the header stripping functionality to work. After the configuration a switch reload is required.
- If **mode tap-aggregation** is configured under interfaces on Cisco Nexus 9300-GX switches, this command is mandatory.
- This command is not required on Cisco Nexus 9300-FX3 platforms.

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 3: 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 4: 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.
- A few guidelines and limitations for the **hardware acl tap-agg** command include:
  - This command is required on Cisco Nexus 9300-GX platform switches for the header stripping functionality to work. After the configuration a switch reload is required.
  - If **mode tap-aggregation** is configured under interfaces on Cisco Nexus 9300-GX switches, this command is mandatory.
  - This command is not required on Cisco Nexus 9300-FX3 platforms.

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

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

- Beginning from Cisco NX-OS Release 10.2(1)F, **feature tap-aggregation** must be enabled for all Tap Aggregation and stripping functions.
- 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 Header Stripping for Nexus Data Broker.
- Beginning with Cisco NX-OS Release 10.3(2)F, EoMPLS label stripping is also supported on Cisco Nexus 9300-FX ToR switches.
- Cisco Nexus 9700-EX and 9700-FX line cards do not support MPLS 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.
- MPLS strip is only supported for Layer 3 packets under the MPLS label stack.
- 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.

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

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 Header Stripping 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

Enable 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.

Procedure

Step 1 Enter global configuration mode using the command **configure terminal**

Example:

```
switch# configure terminal
                        switch(config)#
```

Step 2 Enable TAP aggregation for Cisco Nexus 9700-EX and 9700-FX line cards using the command [**no**] **hardware acl tap-agg**

Example:

```
switch(config)# hardware acl tap-agg
```

This command is also needed on Cisco Nexus 9300-GX and 9300-GX2 platform switches and may require reload.

Step 3 Copy the running configuration to the startup configuration using the command **copy running-config startup-config**

Example:

```
switch(config)# copy running-config startup-config
```

This is an optional step.

Configure TAP Aggregation Policy

Use this task to configure a TAP aggregation policy on an IP or MAC ACL, allowing you to redirect and monitor network traffic as required.

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.

Procedure

Step 1 Enter global configuration mode using the command **configure terminal**

Example:

```
switch# configure terminal
 switch(config)#
```

Step 2 Configure CLIs related to tap-aggregation using the command **feature tap-aggregation**

Example:

```
switch(config)# feature tap-aggregation
 switch(config)#
```



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.

Step 4 Start recording statistics for how many packets are permitted or denied by each entry using the command **statistics per-entry**

Example:

```
switch(config-acl)# statistics per-entry
```

This is an optional step.

Step 5 Create an IP or MAC ACL rule that permits traffic to be redirected per its conditions using the command [**no**] **permit** *protocol source destination* **redirect** *interfaces*

Example:

```
switch(config-acl)# permit ip any any redirect ethernet1/8
```

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 Display all IPv4 or MAC ACLs or a specific IPv4 or MAC ACL using 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
```

This is an optional step.

Step 7 Copy the running configuration to the startup configuration using the command **copy running-config startup-config**

Example:

```
switch(config-acl)# copy running-config startup-config
```

This is an optional step.

Attach TAP Aggregation Policy to Interface

Use this task to configure TAP aggregation policies on IPv4 and MAC ACLs, and to apply or attach these policies to interfaces as required for traffic monitoring and redirection.

Procedure

Step 1 Configure a TAP aggregation policy on an IPv4 ACL.

Example:

```
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
```

Step 2 Apply a TAP aggregation policy to an IPv4 ACL with UDF-based match.

Example:

```
switch# configure terminal
 switch(config)# feature tap-aggregation
 switch(config)# ip access-list tap_agg
 switch(config-acl)# statistics per-entry
 switch(config-acl)# 10 permit ip any any redirect Ethernet1/1
 switch(config-acl)# 20 deny ip any any
 switch# configure terminal
 switch(config)# interface Ethernet1/4
 switch(config-if)# ip port access-group tap_agg in
 switch(config-if)# switchport
 switch(config-if)# no shutdown
```

Step 3 Configure a TAP aggregation policy on a MAC ACL.

Example:

```
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
```

Step 4 Attach a TAP aggregation policy to a Layer 2 interface.

Example:

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

Configure Selective Q-in-Q with Provider VLANs

Before you begin

You must configure provider VLANs

Procedure

Step 1 Enter global configuration mode using the command **configure terminal**

Example:

```
switch# configure terminal
```

Step 2 Enter interface configuration mode for the interface connected to the service provider network using the command **interface** *interface-id*

Example:

```
switch(config)#
interface Ethernet1/1
```

You can enter a physical interface or an EtherChannel port channel.

Step 3 Set the interface as a Layer 2 switching port using the command **switchport** and set the interface as a Layer 2 trunk port using the command **switchport mode trunk**

Example:

```
switch(config if)#
switchport
```

Example:

```
switch(config-if)#
switchport mode trunk
```

Step 4 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 5 Set the allowed VLANs for the trunk interface using the command **switchport trunk allowed vlan** *vlan_list*

Example:

```
switch(config-if)# switchport trunk allowed vlan 300
```

Step 6 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 7 Allow the attachment of an ACL with the tap aggregation policy to the interface using the command **mode tap-aggregation**

Example:

```
switch(config-if)# mode tap-aggregation
                        switch(config-if)# no shutdown
```

This is an optional step.

Step 8 Copy the running configuration to the startup configuration using the command **copy running-config startup-config** and verify the mapping configuration using the command switch(config-if)# **show interfaces** *interface-id* **vlan mapping**

Example:

```
switch(config-if)# copy running-config startup-config
```

This is an optional step.

Step 9 Exit the configuration mode using the command switch(config-if)# **exit**

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)#
```

What to do next

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.

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.

Command	Purpose
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

Enable MPLS Stripping

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 Attach TAP Aggregation Policy to Interface, on page 12 .

You can enable MPLS stripping globally.

Procedure

Step 1 Enter global configuration mode using the command **configure terminal**

Example:

```
switch# configure terminal
 switch(config)#
```

Step 2 Globally enable MPLS stripping using the command [**no**] **mpls strip**

Example:

```
switch(config)# mpls strip
```

The **no** form of this command disables MPLS stripping.

Step 3 Enable VLAN tagging on the packets coming from the redirect port using the command [**no**] **mpls strip mode dot1q**

Example:

```
switch(config)# mpls strip mode dot1q
```

The VLAN that needs to be tagged must be specified in the ingress port.

Step 4 Copy the running configuration to the startup configuration using the command **copy running-config startup-config**

Example:

```
switch(config)# copy running-config startup-config
```

Configure Incoming Port for VLAN Tag

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

Procedure

Step 1 Enter global configuration mode using the command **configure terminal**

Example:

```
switch# configure terminal
 switch(config)#
```

Step 2 Enter interface configuration mode for the specified interface using the command **interface** *type slot/port*

Example:

```
switch(config)# interface ethernet 1/26
 switch(config-if)#
```

Step 3 Change a Layer 3 interface to a Layer 2 interface using the command **switchport**

Example:

```
switch(config-if)# switchport
```

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 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 Copy the running configuration to the startup configuration using the command **copy running-config startup-config**

Example:

```
switch(config-if)# copy running-config startup-config
```

This is an optional step.

Add and Delete 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.

Procedure

Step 1 Enter global configuration mode using the command **configure terminal**

Example:

```
switch# configure terminal
 switch(config)#
```

Step 2 Add the specified static MPLS label using the command **mpls strip label** *label*

Example:

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

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 Copy the running configuration to the startup configuration using the command **copy running-config startup-config**

Example:

```
switch(config)# copy running-config startup-config
```

This is an optional step.

Configure Destination MAC Address

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

Procedure

Step 1 Enter global configuration mode using the command **configure terminal**

Example:

```
switch# configure terminal
   switch(config)#
```

Step 2 Specify the destination MAC address for egress frames that are stripped of their headers using the command **mpls strip dest-mac** *mac-address*

Example:

```
switch(config)# mpls strip dest-mac 1.1.1
```

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 Copy the running configuration to the startup configuration using the command **copy running-config startup-config**

Example:

```
switch(config)# copy running-config startup-config
```

This is an optional step.

Configure MPLS Label Aging

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

Procedure

Step 1 Enter global configuration mode using the command **configure terminal**

Example:

```
switch# configure terminal
 switch(config)#
```

Step 2 Specify the amount of time in seconds after which dynamic MPLS labels age out using the command **mpls strip label-age** *age*

Example:

```
switch(config)# mpls strip label-age 300
```

The range is from 61 to 31622400.

Step 3 Copy the running configuration to the startup configuration using the command **copy running-config startup-config**

Example:

```
switch(config)# copy running-config startup-config
```

This is an optional step.

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

This topic provides references to related documents and topics for further information.

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)	Configuration
Switched port analyzer (SPAN)	Switched Port Analyzer
Troubleshooting	*Cisco Nexus 9000 Series NX-OS Troubleshooting Guide*