Control Plane Traffic

About Control Plane Policing

Control plane policing (CoPP) protects the control plane, which ensures network stability, reachability, and packet delivery.

This feature allows specification of parameters, for each protocol that can reach the control processor to be rate-limited using a policer. The policing is applied to all traffic destined to any of the IP addresses of the router or Layer 3 switch. A common attack vector for network devices is the denial-of-service (DoS) attack, where excessive traffic is directed at the device interfaces.

The Cisco Application Centric Infrastructure (ACI) leaf and spine switch NX-OS provides CoPP to prevent DoS attacks from impacting performance. Such attacks, which can be perpetrated either inadvertently or maliciously, typically involve high rates of traffic destined to the supervisor module of a Cisco ACI leaf and spine switch CPU or CPU itself.

The supervisor module of Cisco ACI leaf and spine switch switches divides the traffic that it manages into two functional components or planes:

- **Data plane**—Handles all the data traffic. The basic functionality of a Cisco NX-OS device is to forward packets from one interface to another. The packets that are not meant for the switch itself are called the transit packets. These packets are handled by the data plane.
- **Control plane**—Handles all routing protocol control traffic. These protocols, such as the Border Gateway Protocol (BGP) and the Open Shortest Path First (OSPF) Protocol, send control packets between devices. These packets are destined to router addresses and are called control plane packets.

The Cisco ACI leaf and spine switch supervisor module has a control plane and is critical to the operation of the network. Any disruption or attacks to the supervisor module will result in serious network outages. For example, excessive traffic to the supervisor module could overload and slow down the performance of the entire Cisco ACI fabric. Another example is a DoS attack on the Cisco ACI leaf and spine switch supervisor module that could generate IP traffic streams to the control plane at a very high rate, forcing the control plane to spend a large amount of time in handling these packets and preventing the control plane from processing genuine traffic.

Examples of DoS attacks are as follows:

- Internet Control Message Protocol (ICMP) echo requests
- IP fragments
- TCP SYN flooding

These attacks can impact the device performance and have the following negative effects:

- Reduced service quality (such as poor voice, video, or critical applications traffic)
- High route processor or switch processor CPU utilization
- Route flaps due to loss of routing protocol updates or keepalives
- Processor resource exhaustion, such as the memory and buffers
- Indiscriminate drops of incoming packets

Note Cisco ACI leaf and spine switches are by default protected by CoPP with default settings. This feature allows for tuning the parameters on a group of nodes based on customer needs.

Control Plane Protection

To protect the control plane, the Cisco NX-OS running on Cisco ACI leaf and spine switches segregates different packets destined for the control plane into different classes. Once these classes are identified, the Cisco NX-OS device polices the packets, which ensures that the supervisor module is not overwhelmed.

Control Plane Packet Types:

Different types of packets can reach the control plane:

- **Receive Packets**—Packets that have the destination address of a router. The destination address can be a Layer 2 address (such as a router MAC address) or a Layer 3 address (such as the IP address of a router interface). These packets include router updates and keepalive messages. Multicast packets can also be in this category where packets are sent to multicast addresses that are used by a router.
- **Exception Packets**—Packets that need special handling by the supervisor module. For example, if a destination address is not present in the Forwarding Information Base (FIB) and results in a miss, the supervisor module sends an ICMP unreachable packet back to the sender. Another example is a packet with IP options set.
- **Redirect Packets**—Packets that are redirected to the supervisor module. Features such as Dynamic Host Configuration Protocol (DHCP) snooping or dynamic Address Resolution Protocol (ARP) inspection redirect some packets to the supervisor module.
- **Glean Packets**—If a Layer 2 MAC address for a destination IP address is not present in the FIB, the supervisor module receives the packet and sends an ARP request to the host.

All of these different packets could be maliciously used to attack the control plane and overwhelm the Cisco ACI fabric. CoPP classifies these packets to different classes and provides a mechanism to individually control the rate at which the Cisco ACI leaf and spine switch supervisor module receives these packets.

Classification for CoPP:

For effective protection, the Cisco ACI leaf and spine switch NX-OS classifies the packets that reach the supervisor modules to allow you to apply different rate controlling policies based on the type of the packet. For example, you might want to be less strict with a protocol packet such as Hello messages but more strict with a packet that is sent to the supervisor module because the IP option is set.

Available Protocols:

Protocol	Description
Glean	With this protocol, when the bridge domain is in proxy mode, unknown unicast traffic received by the leaf switch is sent to the hardware proxy (the spine switch). The spine switch changes the eth-type of the packet to a special eth-type (0xfff2). When these packets reach the leaf switches through the fabric ports, the packets are classified under glean. The packets are sent to the leaf switch's CPU, and the leaf switch's CPU generates an ARP request for the connected external devices.
ToR Glean	ToR glean activates when an endpoint moves or is cleared because of link flap and does not update the source leaf switch's remote IP address endpoint entry. A packet egresses the source leaf switch with the destination leaf switch's TEP address. On the destination leaf switch, because of the missing local IP address entry, the packet gets sent to the leaf switch CPU to generate an ARP request for those IP addresses. These packets are classified under ToR glean.

Rate Controlling Mechanisms:

Once the packets are classified, the Cisco ACI leaf and spine switch NX-OS has different mechanisms to control the rate at which packets arrive at the supervisor module.

You can configure the following parameters for policing:

- **Committed information rate (CIR)**—Desired bandwidth, specified as a bit rate or a percentage of the link rate.
- **Committed burst (BC)**—Size of a traffic burst that can exceed the CIR within a given unit of time and not impact scheduling.

Default Policing Policies:

When the Cisco ACI leaf and spine switch is bootup, the platform setup pre-defined CoPP parameters for different protocols are based on the tests done by Cisco.

Guidelines and Limitations for CoPP

CoPP has the following configuration guidelines and limitations:

- We recommend that you use the default CoPP policy initially and then later modify the CoPP policies based on the data center and application requirements.

- Customizing CoPP is an ongoing process. CoPP must be configured according to the protocols and features used in your specific environment as well as the supervisor features that are required by the server environment. As these protocols and features change, CoPP must be modified.
- We recommend that you continuously monitor CoPP. If drops occur, determine if CoPP dropped traffic unintentionally or in response to a malfunction or attack. In either event, analyze the situation and evaluate the need to modify the CoPP policies.
- You must ensure that the CoPP policy does not filter critical traffic such as routing protocols or interactive access to the device. Filtering this traffic could prevent remote access to the Cisco ACI Leaf/Spine and require a console connection.
- Do not mis-configure CoPP pre-filter entries. CoPP pre-filter entries might impact connectivity to multi-pod configurations, remote leaf switches, and Cisco ACI Multi-Site deployments.
- You can use the APIC UI to be able to tune the CoPP parameters.
- Per interface per protocol is only supported on Leaf switches.
- FEX ports are not supported on per interface per protocol.
- For per interface per protocol the supported protocols are; ARP, ICMP, CDP, LLDP, LACP, BGP, STP, BFD, and OSPF.
- The TCAM entry maximum for per interface per protocol is 256. Once the threshold is exceeded a fault will be raised.

Configuring CoPP Using the APIC GUI

Procedure

Step 1 On the menu bar, click **Fabric** > **External Access Policies**.

Step 2 In the **Navigation** pane, expand **Policies** > **Switch** > **CoPP Leaf**, right click **Create Profiles for CoPP To Be Applied At The Leaf Level** dialog box to perform the following actions in the **Create Profiles for CoPP To Be Applied At The Leaf Level** dialog box:

a) In the **Name** field, add a policy name.
b) In the **Type of Profile** field, select the profile type.

Note Select **CoPP has custom values** if you wish to set each protocol separately. If you do not select a profile type then default values are applied.

c) Click **Submit** to create the policy.

Step 3 In the **Navigation** pane, expand **Switches** > **Leaf Switches** > **Policy Groups**, right click **Create Access Switch Policy Group** dialog box to perform the following actions in the **Create Access Switch Policy Group** dialog box:

a) In the **Name** field, add a policy name.
b) In the **COPP Leaf Policy** field, select the policy previously created.
c) Click **Submit**.

Step 4 In the **Navigation** pane, expand **Switches** > **Leaf Switches** > **Profiles**, right click **Create Leaf Profile** dialog box to perform the following actions in the **Create Leaf Profile** dialog box:

a) In the **Name** field, add a profile name.
b) Expand the **Leaf Selectors** table, add the Leaf information in the **Name** and **Blocks** fields, and select the **Policy Group** previously created.
c) Click **Next** and **Finish** to complete CoPP configuration.

Configuring CoPP Using the Cisco NX-OS CLI

Procedure

Step 1 Configure a CoPP leaf profile:

Example:

```
# configure copp Leaf Profile
apic1(config)# policy-map type control-plane-leaf leafProfile
apic1(config-pmap-copp-leaf)# profile-type custom
apic1(config-pmap-copp-leaf)# set arpRate 786
# create a policy group to be applied on leaves
apic1(config)# template leaf-policy-group coppForLeaves
apic1(config-leaf-policy-group)# copp-aggr leafProfile
apic1(config-leaf-policy-group)# exit
# apply the leaves policy group on leaves
apic1(config)# leaf-profile applyCopp
apic1(config-leaf-profile)# leaf-group applyCopp
apic1(config-leaf-group)# leaf 101-102
apic1(config-leaf-group)# leaf-policy-group coppForLeaves
```

Step 2 Configure a CoPP Spine profile:

Example:

```
# configure copp Spine Profile
apic1(config)# policy-map type control-plane-spine spineProfile
apic1(config-pmap-copp-spine)# profile-type custom
apic1(config-pmap-copp-spine)# set arpRate 786
# create a policy group to be applied on spines
apic1(config)# template leaf-policy-group coppForSpines
apic1(config-spine-policy-group)# copp-aggr spineProfile
apic1(config-spine-policy-group)# exit
# apply the spine policy group on spines
apic1(config)# spine-profile applyCopp
apic1(config-spine-profile)# spine-group applyCopp
apic1(config-spine-group)# spine 201-202
apic1(config-spine-group)# spine-policy-group coppForSpines
```

Configuring CoPP Using the REST API

Procedure

Step 1 Configure a CoPP leaf profile:

Example:

```
<!-- api/node/mo/uni/.xml -->
<infraInfra>
  <coppLeafProfile type="custom" name="mycustom">                    <!-- define copp leaf
profile -->
    <coppLeafGen1CustomValues bgpBurst="150" bgpRate="300"/>
  </coppLeafProfile>
  <infraNodeP name="leafCopp">
    <infraLeafS name="leafs" type="range">
      <infraNodeBlk name="leaf1" from_="101" to_="101"/>
      <infraNodeBlk name="leaf3" from_="103" to_="103"/>
      <infraRsAccNodePGrp tDn="uni/infra/funcprof/accnodepgrp-myLeafCopp"/>
    </infraLeafS>
  </infraNodeP>
  <infraFuncP>
    <infraAccNodePGrp name="myLeafCopp">
      <infraRsLeafCoppProfile tnCoppLeafProfileName="mycustom"/>   <!-- bind copp leaf
policy to leaf </infraAccNodePGrp>
profile -->
  </infraFuncP>
</infraInfra>
```

Step 2 Configure a CoPP spine profile:

Example:

```
<!-- api/node/mo/uni/.xml -->
<infraInfra>
  <coppSpineProfile type="custom" name="mycustomSpine">             <!-- define copp leaf
profile -->
    <coppSpineGen1CustomValues bgpBurst="150" bgpRate="300"/>
  </coppSpineProfile>
  <infraSpineP name="spineCopp">
    <infraSpineS name="spines" type="range">
      <infraNodeBlk name="spine1" from_="104" to_="104"/>
      <infraRsSpineAccNodePGrp tDn="uni/infra/funcprof/spaccnodepgrp-mySpineCopp"/>
    </infraSpineS>
  </infraSpineP>
  <infraFuncP>
    <infraSpineAccNodePGrp name="mySpineCopp">
     <infraRsSpineCoppProfile tnCoppSpineProfileName="mycustomSpine"/> <!-- bind copp spine
 policy to
    </infraSpineAccNodePGrp>                                                spine profile
 -->
  </infraFuncP>
</infraInfra>
```

Viewing CoPP Statistics Using the GUI

Fine tuning CoPP requires knowing the number of packets dropped/allowed by a given protocol on a given node. The information can be viewed in the GUI using the procedure below:

Procedure

On the menu bar, click **Fabric** > **Inventory** > **Pod***number* > **Node***name* > **Control Plane Statistics** > **default**, select from the list of classes to configure the statistics display format.

You can collect statistics about the number of packets allowed or dropped by CoPP.

Configuring Per Interface Per Protocol CoPP Policy Using the APIC GUI

Procedure

Step 1 On the menu bar, click **Fabric** > **External Access Policies**.

Step 2 In the **Navigation** pane, expand **Policies** > **Interface** > **CoPP Interface**, right click **Create Per Interface Per Protocol CoPP Policy** dialog box to perform the following actions in the **Create Per Interface Per Protocol CoPP Policy** dialog box:

a) In the **Name** field, add a policy name.
b) Expand the **CoPP policy Protocol** table, and enter the protocol name, type, rate, and burst information. Click **Update** and **Submit**.

Step 3 In the **Navigation** pane, expand **Interfaces** > **Leaf Interfaces** > **Policy Groups** > **Create Leaf Access Port Policy Group**, right click **Create Leaf Access Port Policy Group** dialog box to perform the following actions in the **Create Leaf Access Port Policy Group** dialog box:

a) In the **Name** field, add a policy name.
b) In the **COPP Leaf Policy** field, select the policy previously created.
c) Click **Submit**.

Step 4 In the **Navigation** pane, expand **Interfaces** > **Leaf Interfaces** > **Profiles** > **Leaf Profiles**, right click **Create Leaf Interface Profile** dialog box to perform the following actions in the **Create Leaf Interface Profile** dialog box:

a) In the **Name** field, add a profile name.
b) Expand the **Interface Selectors** table, add the interface information in the **Name** and **Interface IDs** fields, and select the **Interface Policy Group** previously created.
c) Click **Ok** and **Submit** to complete Per Interface Per Protocol CoPP configuration.

Configuring Per Interface Per Protocol CoPP Policy Using the NX-OS Style CLI

Procedure

Step 1 Define the CoPP class map and policy map:

Example:

```
(config)# policy-map type control-plane-if <name>
        (config-pmap-copp)# protocol bgp bps <value>
        (config-pmap-copp)# protocol ospf bps <value>
```

Step 2 Applying the configuration to an interface on the leaf:

Example:

```
(config)# leaf 101
        (config-leaf)# int eth 1/10
        (config-leaf-if)# service-policy type control-plane-if output<name>
```

Configuring CoPP Per Interface Per Protocol Using REST API

Procedure

Configure a CoPP per interface per protocol:

Example:

```
<polUni>
    <infraInfra>
    <infraNodeP name="default">
        <infraLeafS name="default" type="range">
            <infraNodeBlk name="default" to_="101" from_="101"/>
        </infraLeafS>
        <infraRsAccPortP tDn="uni/infra/accportprof-default"/>
    </infraNodeP>
    <infraAccPortP name="default">
        <infraHPortS name="regularPorts" type="range">
            <infraPortBlk name="blk1" toPort="7" fromPort="1" toCard="1" fromCard="1"/>
                <infraRsAccBaseGrp tDn="uni/infra/funcprof/accportgrp-copp"/>
        </infraHPortS>
    </infraAccPortP>

    <infraFuncP>
        <infraAccPortGrp name="copp">
            <infraRsCoppIfPol tnCoppIfPolName="pc"/>
        </infraAccPortGrp>
    </infraFuncP>

    <coppIfPol name = "pc" >
        <coppProtoClassP name = "test" matchProto="lldp,arp" rate="505" burst = "201"/>
        <coppProtoClassP name = "test1" matchProto="bgp" rate="500" burst = "200" />
    </coppIfPol>
</infraInfra>
</polUni>
```

About CoPP Prefilters

In Cisco Application Centric Infrastructure (ACI), you can use the control plane policing (CoPP) prefilter feature to filter control packets sent to the CPU. A CoPP prefilter is the same as an inftrastructure access control list (iACL).

Before you use this feature, keep in mind the following key points:

1. This feature works leaf switch-wise or spine switch-wise, not per interface, nor per-L3Out.
2. This feature takes effect across VRF instances, meaning that the filters that you define are not specific to a VRF instance. If you enable a CoPP prefilter and you do not specifically allow ICMP traffic in the

configuration of the CoPP prefilter, ICMP traffic sent to the bridge domains of any VRF instance of a given leaf switch is dropped.

3. A CoPP prefilter is configured as a permit-list.
4. This feature is activated by entering the first filtering rule. This means that if you do not have any filtering rules configured, everything is allowed. As soon as you enter the first rule, then everything else is dropped except the traffic that you allow in the filtering rules. This means that all the IPv4/IPv6 control plane traffic by default is denied unless you add it to the permit-list.
5. The filter configuration allows you to enter protocols/DIP/SIP/Protocol/L4 port/L4 port range. You can enter the source and destination IP address of the traffic.
6. You must also allow underlay protocols that are not implicitly allowed. For example, you must allow BGP, otherwise the infra BGP sessions to the leaf or spine switch go down. As another example, you must allow OSPF for remote leaf reachability if you enable this feature on the remote leaf switch.
7. Because of point #6, if you configure a CoPP prefilter on leaf or spine switches of a single POD, you must make sure BGP and DHCP traffic is allowed. If the spine switch is also connected to an IPN/ISN, you must consider allowing OSPF.
8. Because of point #6, in Cisco ACI Multi-Pod, Cisco ACI Multi-Site or Cisco Nexus Dashboard Orchestrator, GOLF, or a remote leaf switch, you must add BGP, DHCP, and OSPF to the permit-list for infra connectivity.
9. Enabling the feature does not disconnect the leaf switch from the fabric because Cisco Application Policy Infrastructure Controller (APIC) traffic is automatically allowed. But, be aware that unless you specifically add BGP to the permit-listed, enabling this feature disconnects the infa BGP session to the leaf switch.
10. The following things are automatically allowed: COOP traffic, vPC control plane traffic, protocols such as LACP/LLDP/CDP, ARP, and Neighbour Discovery packets (RS/RA/NS/NA).
11. ICMP, IGMP, and any other protocol must be specifically allowed. If you enable a CoPP prefilter and you want to make sure that servers can ping the bridge domain subnet IP address, you must make sure ICMP is allowed.
12. There is no support for an ICMP sub-type to allow only ICMP replies or requests. Enabling ICMP enables both.

Supported Platforms

This section lists the supported platforms for the CoPP prefilter feature.

Supported leaf switches:

- N9K-C93108TC-EX
- N9K-C93108TC-FX
- N9K-C93108YC-FX
- N9K-C93180LC-EX
- N9K-C93180YC-EX
- N9K-C9348GC-FXP

Supported spine switches:

- N9K-C92300YC
- N9K-C92304QC
- N9K-C9232C
- N9K-C9236C
- N9K-C9272Q
- N9K-C9364C
- N9K-C9508-FM-2
- N9K-C9516-FM-E2

Limitations

- Only Ethernet type IPv4 or IPv6 packets can be matched in the egress TCAM. ARP and ND packets are not matched.
- A total of 128 (wide key) entries can be included in the allowed list. However, some entries are reserved for internal use.

Configuring a CoPP Prefilter, Policy Group, and Profile Using the GUI

Configuring a CoPP Prefilter Using the Cisco APIC GUI

This section explains how to configure a CoPP prefilter at the leaf level and the spine level using the Cisco APIC GUI.

Before you begin

Access to the APIC GUI

Procedure

Step 1 Click **Fabric** > **External Access Policies**.

Step 2 From the **Navigation** pane, click **Policies** > **Switch**.
The **CoPP Pre-Filter for Leaf** and **CoPP Pre-Filter for Spine** nodes appear in the **Navigation** pane.

Step 3 From the **Navigation** pane, choose between the following options:

- **CoPP Pre-Filter for Leaf**–To create a CoPP prefilter for a leaf switch, right-click on **CoPP Pre-Filter for Leaf** and choose **Create Profiles for CoPP Pre-Filter To Be Applied At The Leaf Level**.
- **CoPP Pre-Filter for Spine**–To create a CoPP prefilter for a spine switch, right-click on **CoPP Pre-Filter for Spine** and choose **Create Profiles for CoPP Pre-Filter To Be Applied At The Spine Level**

The respective CoPP prefilter dialog appears.

Step 4 Enter the appropriate values in the dialog fields.

Note For information about the fields in the dialog, click the help icon to display the Cisco APIC help file.

Step 5 When finished, click **Submit**.

What to do next

Configure a policy group.

Configuring a Leaf Policy Group Using the GUI

This section explains how to create a policy group.

Before you begin

Access to a Cisco APIC GUI.

Procedure

Step 1 Click **Fabric** > **External Access Policies**.

Step 2 From the **Navigation** pane, click **Switches** > **Leaf Switches**.
The **Policy Groups** node appears in the **Navigation** pane.

Step 3 From the **Navigation** pane, **Policy Groups**–To create a leaf policy group, right-click on **Policy Groups** and choose **Create Access Switch Policy Group**.

The respective policy group dialog appears.

Step 4 From the policy group dialog, enter a name in the **Name** field and click the drop-down arrow of the policy type you want to apply. Any configured policies for the chosen policy type will appear in the drop-down list.

Note For information about the fields in the dialog, click the help icon to display the Cisco APIC help file.

Step 5 When finished, click **Submit**.

What to do next

Configure a profile.

Configuring a Leaf Profile Using the GUI

This section explains how to create a profile.

Before you begin

You should have a configured policy group.

Procedure

Step 1 Click **Fabric** > **External Access Policies**.

Step 2 From the **Navigation** pane, click **Switches** > **Leaf Switches** > **Profiles**.
The **Leaf Profiles** node appears in the **Navigation** pane.

Step 3 From the **Navigation** pane, **Profiles**–To create a profile for a leaf switch, right-click on **Profiles** and choose **Create Leaf Profile**.

The respective profile dialog appears.

Step 4 From the profile dialog, enter a name in the **Name** field and click the **+** to enter the selector information. Click **Update** when finished.

After clicking **Update**, you return to the profile dialog.

Step 5 Click **Next** to enter the interface selector profile information.

Note For information about the fields in the dialog, click the help icon to display the Cisco APIC help file.

Step 6 When finished, click **Finish**.

Configuring a CoPP Prefilter Using the CLI

Configuring the CoPP Prefilter for a Leaf Switch Using the CLI

This section explains how to configure a CoPP prefilter policy and policy group then associate a switch policy group with a switch profile using the CLI.

Procedure

Step 1 Switch# **configure terminal**

Enters global configuration mode.

Step 2 Switch(config)# **template control-plane-policing-prefilter-leaf** <*name*>

Creates a CoPP prefilter profile for a leaf switch.

Step 3 Switch (config-control-plane-policing-prefilter-leaf)# **permit proto** { **tcp** | **udp** | **eigrp** | **unspecified** | **icmp** | **icmpv6** | **egp** | **igp** | **l2tp** | **ospf** | **pim** }

Permits the specified IP protocol.

Step 4 Switch (config-control-plane-policing-prefilter-leaf)#**exit**

Enters global configuration mode.

Step 5 Switch(config)# **template leaf-policy-group** <*name*>

Creates a CoPP prefilter policy group leaf switches.

Step 6 Switch(config-leaf-policy-group)# **control-plane-policing-prefilter** <*name*>

Associates a leaf policy group with the CoPP prefilter policy.

Step 7 Switch(config-leaf-policy-group)# **exit** <*name*>

Enters global configuration mode.

Step 8 Switch(config)# **leaf-profile** <*name*>

Creates a leaf profile.

Step 9 Switch(config-leaf-profile)# **leaf-group** <*name*>

Associates a leaf group with a leaf profile.

Step 10 Switch(config-leaf-group)# **leaf-policy-group** <*name*>

Associates a leaf policy group with a leaf group.

Configuring the CoPP Prefilter for a Spine Switch Using the CLI

This section explains how to configure a CoPP prefilter policy and policy group then associate a switch policy group with a switch profile using the CLI.

Procedure

Step 1 Switch# **configure terminal**

Enters global configuration mode.

Step 2 Switch(config)# **template control-plane-policing-prefilter-spine** <*name*>

Creates a CoPP prefilter profile for a spine switch.

Step 3 Switch (config-control-plane-policing-prefilter-spine)# **permit proto** { **tcp** | **udp** | **eigrp** | **unspecified** | **icmp** | **icmpv6** | **egp** | **igp** | **l2tp** | **ospf** | **pim** }

Permits the specified IP protocol.

Step 4 Switch (config-control-plane-policing-prefilter-spine)#**exit**

Enters global configuration mode.

Step 5 Switch(config)# **template spine-policy-group** <*name*>

Creates a CoPP prefilter policy group spine switches.

Step 6 Switch(config-spine-policy-group)# **control-plane-policing-prefilter** <*name*>

Associates a spine policy group with the CoPP prefilter policy.

Step 7 Switch(config-spine-policy-group)# **exit** <*name*>

Enters global configuration mode.

Step 8 Switch(config)# **spine-profile** <*name*>

Creates a spine profile.

Step 9 Switch(config-spine-profile)# **spine-group** <*name*>

Associates a spine group with a spine profile.

Step 10 Switch(config-spine-group)# **spine-policy-group** <*name*>

Associates a spine policy group with a spine group.

Configuring a CoPP Prefilter Using the REST API

Configuring a CoPP Prefilter Policy for a Leaf Switch Using the REST API

This section explains how to configure a CoPP prefilter policy for a leaf switch using the REST API.

Procedure

Step 1 Create a switch policy for CoPP Prefilter with entries the allowed list.

```
<iaclLeafProfile descr="" dn="uni/infra/iaclspinep-spine_icmp" name="COPP_PreFilter_BGP_Config
 " ownerKey="" ownerTag="">
<iaclEntry dstAddr="0.0.0.0/0" dstPortFrom="179" dstPortTo="179" ipProto="tcp" name="bgp"
nameAlias="" srcAddr="0.0.0.0/0" srcPortFrom="179" srcPortTo="179"/>
</iaclLeafProfile>
```

Step 2 Create a switch policy group with CoPP prefilter policies.

```
<infraAccNodePGrp descr="" dn="uni/infra/funcprof/accnodepgrp-COPP_PreFilter_BGP_Config "
name="COPP_PreFilter_BGP_Config" nameAlias="" ownerKey="" ownerTag="">
<infraRsIaclLeafProfile tnIaclLeafProfileName="COPP_PreFilter_BGP_Config"/>
</infraAccNodePGrp>
```

Step 3 Associate switch policy group to switch profiles.

```
<infraNodeP descr="" dn="uni/infra/nprof-leafP-103" name="leafP-103" nameAlias="" ownerKey=""
 ownerTag="">
<infraLeafS descr="" name="103_Sel" nameAlias="" ownerKey="" ownerTag="" type="range">
<infraRsAccNodePGrp tDn="uni/infra/funcprof/accnodepgrp-COPP_PreFilter_BGP_Config"/>
<infraNodeBlk descr="" from_="103" name="nblk1" nameAlias="" to_="103"/>
</infraLeafS>
</infraNodeP>
```

Configuring a CoPP Prefilter Policy for a Spine Using the REST API

This section explains how to configure a CoPP prefilter policy for a spine switch using the REST API.

Procedure

Step 1 Create a switch policy for CoPP Prefilter with entries the allowed list.

```
<iaclSpineProfile descr="" dn="uni/infra/iaclspinep-spine_icmp"
name="COPP_PreFilter_OSPF_Config" ownerKey="" ownerTag="">
<iaclEntry dstAddr="0.0.0.0/0" dstPortFrom="unspecified" dstPortTo="unspecified"
ipProto="ospfigp" name="" nameAlias="" srcAddr="0.0.0.0/0" srcPortFrom="unspecified"
srcPortTo="unspecified"/>
</iaclSpineProfile>
```

Step 2 Create a switch policy group with CoPP prefilter policies.

```
<infraSpineAccNodePGrp descr=""
dn="uni/infra/funcprof/spaccnodepgrp-COPP_PreFilter_OSPF_Config"
name="COPP_PreFilter_OSPF_Config" nameAlias="" ownerKey="" ownerTag="">
<infraRsIaclSpineProfile tnIaclSpineProfileName="COPP_PreFilter_OSPF_Config"/>
</infraSpineAccNodePGrp>
```

Step 3 Associate switch policy group to switch profiles.

```
<infraSpineP descr="" dn="uni/infra/spprof-204" name="204" nameAlias="" ownerKey=""
ownerTag="">
<infraSpineS descr="" name="204" nameAlias="" ownerKey="" ownerTag="" type="range">
<infraRsSpineAccNodePGrp tDn="uni/infra/funcprof/spaccnodepgrp-COPP_PreFilter_OSPF_Config"/>
<infraNodeBlk descr="" from_="204" name="nodeblock1" nameAlias="" to_="204"/>
</infraSpineS>
<infraRsSpAccPortP tDn="uni/infra/spaccportprof-204"/>
</infraSpineP>
```

What to do next