Configuring QoS

QoS for L3Outs

QoS for L3Outs

To configure QoS policies for an L3Out, use the following guidelines:

- To configure the QoS policy to be enforced on the border leaf where the L3Out is located, the VRF instance must be in egress mode (Policy Control Enforcement Direction must be "Egress").
- To enable the QoS policy to be enforced, the VRF Policy Control Enforcement Preference must be "Enforced."
- When configuring the contract governing communication between the L3Out and other EPGs, include the QoS class or target DSCP in the contract or subject.

Note Only configure a QoS class or target DSCP in the contract, not in the external EPG (`l3extInstP`).

- When creating a contract subject, you must choose a QoS priority level. You cannot choose Unspecified.

Configuring QoS for L3Outs Using the REST API

QoS for L3Out is configured as part of the L3Out configuration.

Procedure

Step 1 When configuring the tenant, VRF, and bridge domain, configure the VRF for egress mode (`pcEnfDir="egress"`) with policy enforcement enabled (pcEnfPref="enforced"). Send a post with XML similar to the following example:

Example:

```
<fvTenant  name="t1">
      <fvCtx name="v1" pcEnfPref="enforced" pcEnfDir="egress"/>
        <fvBD name="bd1">
            <fvRsCtx tnFvCtxName="v1"/>
            <fvSubnet ip="44.44.44.1/24" scope="public"/>
            <fvRsBDToOut tnL3extOutName="l3out1"/>
        </fvBD>"/>
</fvTenant>
```

Step 2 When creating the filters and contracts to enable the EPGs participating in the L3Out to communicate, configure the QoS priority.

The contract in this example includes the QoS priority, `level1`, for traffic ingressing on the L3Out. Alternatively, it could define a target DSCP value. QoS policies are supported on either the contract or the subject.

The filter also has the matchDscp="EF" criteria, so that traffic with this specific TAG received by the L3out processes through the queue specified in the contract subject.

Example:

```
<vzFilter name="http-filter">
     <vzEntry  name="http-e" etherT="ip" prot="tcp" matchDscp="EF"/>
</vzFilter>
<vzBrCP name="httpCtrct" prio="level1" scope="context">
     <vzSubj name="subj1">
          <vzRsSubjFiltAtt tnVzFilterName="http-filter"/>
     </vzSubj>
</vzBrCP>
```

Configuring QoS Directly on L3Out Using REST API

This section describes how to configure QoS directly on an L3Out. This is the preferred way of configuring L3Out QoS starting with Cisco APIC Release 4.0(1).

You can configure QoS for L3Out on one of the following objects:

- Switch Virtual Interface (SVI)
- Sub Interface
- Routed Outside

Procedure

Step 1 Configure QoS priorities for a L3Out SVI.

Example:

```
<l3extLIfP descr=""
dn="uni/tn-DT/out-L3_4_2_24_SVI17/lnodep-L3_4_E2_24/lifp-L3_4_E2_24_SVI_19"
           name="L3_4_E2_24_SVI_19" prio="level6" tag="yellow-green">
    <l3extRsPathL3OutAtt addr="0.0.0.0" autostate="disabled" descr="SVI19" encap="vlan-19"
                         encapScope="local" ifInstT="ext-svi" ipv6Dad="enabled" llAddr="::"

                          mac="00:22:BD:F8:19:FF" mode="regular" mtu="inherit"
                          tDn="topology/pod-1/protpaths-103-104/pathep-[V_L3_l4_2-24]"
                          targetDscp="unspecified">
        <l3extMember addr="107.2.1.253/24" ipv6Dad="enabled" llAddr="::" side="B"/>
        <l3extMember addr="107.2.1.252/24" ipv6Dad="enabled" llAddr="::" side="A"/>
    </l3extRsPathL3OutAtt>
    <l3extRsLIfPCustQosPol tnQosCustomPolName="VrfQos006"/>
</l3extLIfP>
```

Step 2 Configure QoS priorities for a sub-interface.

Example:

```
<l3extLIfP dn="uni/tn-DT/out-L4E48_inter_tenant/lnodep-L4E48_inter_tenant/lifp-L4E48"
           name="L4E48" prio="level4" tag="yellow-green">
    <l3extRsPathL3OutAtt addr="210.1.0.254/16" autostate="disabled" encap="vlan-20"
                         encapScope="local" ifInstT="sub-interface" ipv6Dad="enabled"
llAddr="::"
                         mac="00:22:BD:F8:19:FF" mode="regular" mtu="inherit"
                         tDn="topology/pod-1/paths-104/pathep-[eth1/48]"
targetDscp="unspecified"/>
    <l3extRsNdIfPol annotation="" tnNdIfPolName=""/>
    <l3extRsLIfPCustQosPol annotation="" tnQosCustomPolName=" vrfQos002"/>
</l3extLIfP>
```

Step 3 Configure QoS priorities for a routed outside.

Example:

```
<l3extLIfP dn="uni/tn-DT/out-L2E37/lnodep-L2E37/lifp-L2E37OUT"
           name="L2E37OUT" prio="level5" tag="yellow-green">
    <l3extRsPathL3OutAtt addr="30.1.1.1/24" autostate="disabled" encap="unknown"
                         encapScope="local" ifInstT="l3-port" ipv6Dad="enabled"
                         llAddr="::" mac="00:22:BD:F8:19:FF" mode="regular"
                         mtu="inherit" targetDscp="unspecified"
                         tDn="topology/pod-1/paths-102/pathep-[eth1/37]"/>
    <l3extRsNdIfPol annotation="" tnNdIfPolName=""/>
    <l3extRsLIfPCustQosPol tnQosCustomPolName="vrfQos002"/>
</l3extLIfP>
```

CoS Preservation

Preserving 802.1P Class of Service Settings

APIC enables preserving 802.1P class of service (CoS) settings within the fabric. Enable the fabric global QoS policy `dot1p-preserve` option to guarantee that the CoS value in packets which enter and transit the ACI fabric is preserved.

802.1P CoS preservation is supported in single pod and multipod topologies.

In multipod topologies, CoS Preservation can be used where you want to preserve the QoS priority settings of 802.1P traffic entering POD 1 and egressing out of POD 2, but you are not concerned with preserving the CoS/DSCP settings in interpod network (IPN) traffic between the pods. To preserve CoS/DSCP settings when multipod traffic is transitting an IPN, use a DSCP policy. For more information, see Preserving QoS Priority Settings in a Multipod Fabric, on page 5.

Observe the following 801.1P CoS preservation guidelines and limitations:

- The current release can only preserve the 802.1P value within a VLAN header. The DEI bit is not preserved.
- For VXLAN encapsulated packets, the current release will not preserve the 802.1P CoS value contained in the outer header.
- 802.1P is not preserved when the following configuration options are enabled:
  - Multipod QoS (using a DSCP policy) is enabled.
  - Contracts are configured that include QoS.
  - Dynamic packet prioritization is enabled.
  - The outgoing interface is on a FEX.
  - Preserving QoS CoS priority settings is not supported when traffic is flowing from an EPG with isolation enforced to an EPG without isolation enforced.
  - A DSCP QoS policy is configured on a VLAN EPG and the packet has an IP header. DSCP marking can be set at the filter level on the following with the precedence order from the innermost to the outermost:
    - Contract
    - Subject
    - In Term
    - Out Term

Note When specifying vzAny for a contract, external EPG DSCP values are not honored because vzAny is a collection of all EPGs in a VRF, and EPG specific configuration cannot be applied. If EPG specific target DSCP values are required, then the external EPG should not use vzAny.

Preserving QoS CoS Settings Using the REST API

Procedure

Step 1 Enable CoS preservation, using a REST API POST statement, similar to the following:

Example:

```
post https://192.0.20.123/api/node/mo/uni/infra/qosinst-default.xml
<imdata totalCount="1">

<qosInstPol ownerTag="" ownerKey="" name="default" dn="uni/infra/qosinst-default" descr=""
 ctrl="dot1p-preserve"/>

</imdata>
```

Step 2 Disable CoS preservation, using a POST statement, such as the following example, which leaves the `ctrl` property empty:

Example:

```
post https://192.0.20.123/api/node/mo/uni/infra/qosinst-default.xml
<imdata totalCount="1">

<qosInstPol ownerTag="" ownerKey="" name="default" dn="uni/infra/qosinst-default" descr=""
 ctrl=""/>

</imdata>
```

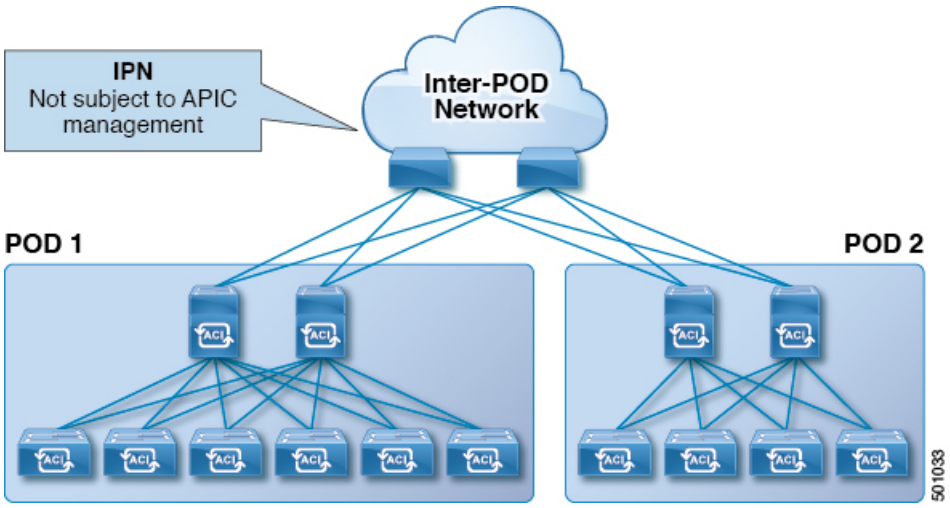
Multipod QoS

Preserving QoS Priority Settings in a Multipod Fabric

This topic describes how to guarantee QoS priority settings in a multipod topology, where devices in the interpod network are not under APIC management, and may modify 802.1P settings in traffic transitting their network.

Note You can alternatively use CoS Preservation where you want to preserve the QoS priority settings of 802.1P traffic entering POD 1 and egressing out of POD 2, but you are not concerned with preserving the CoS/DSCP settings in interpod network (IPN) traffic between the pods. For more information, see Preserving 802.1P Class of Service Settings, on page 3.

Figure 1: Multipod Topology



As illustrated in this figure, traffic between pods in a multipod topology passes through an IPN, which may not be under APIC management. When an 802.1P frame is sent from a spine or leaf switch in POD 1, the devices in the IPN may not preserve the CoS setting in 802.1P frames. In this situation, when the frame reaches a POD 2 spine or leaf switch, it has the CoS level assigned by the IPN device, instead of the level assigned at the source in POD 1. Use a DSCP policy to ensure that the QoS priority levels are preserved in this case.

Configure a DSCP policy to preserve the QoS priority settings in a multipod topology, where there is a need to do deterministic mapping from CoS to DSCP levels for different traffic types, and you want to prevent the devices in the IPN from changing the configured levels. With a DSCP policy enabled, APIC converts the CoS level to a DSCP level, according to the mapping you configure. When a frame is sent from POD 1 (with the PCP level mapped to a DSCP level), when it reaches POD 2, the mapped DSCP level is then mapped back to the original PCP CoS level.

Creating a DSCP Policy Using the REST API

Procedure

Step 1 Configure and enable a DSCP policy with a post, such as the following:

Example:

```
post https://192.0.20.123/api/node/mo/uni/tn-infra/dscptranspol-default.xml

<imdata totalCount="1">

<qosDscpTransPol traceroute="AF43" span="AF42" policy="AF22" ownerTag="" ownerKey=""
name="default"
level3="AF13" level2="AF12" level1="AF11" dn="uni/tn-infra/dscptranspol-default" descr=""
control="AF21" adminSt="enabled"/>

</imdata>
```

Step 2 Disable the DSCP policy with a post such as the following:

Example:

```
post https://192.0.20.123/api/node/mo/uni/tn-infra/dscptranspol-default.xml

<imdata totalCount="1">

<qosDscpTransPol traceroute="AF43" span="AF42" policy="AF22" ownerTag="" ownerKey=""
name="default"
level3="AF13" level2="AF12" level1="AF11" dn="uni/tn-infra/dscptranspol-default" descr=""
control="AF21" adminSt="disabled"/>

</imdata>
```

Translating QoS Ingress Markings to Egress Markings

Translating QoS Ingress Markings to Egress Markings

APIC enables translating the 802.1P CoS field (Class of Service) based on the ingress DSCP value. 802.1P CoS translation is supported only if DSCP is present in the IP packet and dot1P is present in the Ethernet frames.

This functionality enables the ACI Fabric to classify the traffic for devices that classify the traffic based only on the CoS value. It allows mapping the dot1P CoS value based on the ingress dot1P value. It is mainly applicable for Layer 2 packets, which do not have an IP header.

Observe the following 802.1P CoS translation guidelines and limitations:

- Enable the fabric global QoS policy `dot1p-preserve` option.
- 802.1P CoS translation is not supported on external L3 interfaces.
- 802.1P CoS translation is supported only if the egress frame is 802.1Q encapsulated.

802.1P CoS translation is not supported when the following configuration options are enabled:

- Contracts are configured that include QoS.
- The outgoing interface is on a FEX.
- Multipod QoS using a DSCP policy is enabled.
- Dynamic packet prioritization is enabled.
- If an EPG is configured with intra-EPG endpoint isolation enforced.
- If an EPG is configured with allow-microsegmentation enabled.

Translating QoS Ingress Markings to Egress Markings Using the REST API

Create a custom QoS policy and then associate the policy with an EPG.

Before you begin

Create the tenant, application, and EPGs that will consume the custom QoS policy. The example creates the `vrfQos001` custom QoS policy and associates it with the `ep2` EPG, that will consume it.

Procedure

Step 1 Create a custom QoS policy by sending a post with XML such as the following example:

Example:

```
<qosCustomPol name="vrfQos001" dn="uni/tn-t001/qoscustom-vrfQos001">
    <qosDscpClass to="AF31" targetCos="6"
        target="unspecified" prio="unspecified" from="AF23"/>
    <qosDot1PClass to="1" targetCos="6" target="unspecified"
       prio="unspecified" from="0"/>
</qosCustomPol>
```

Step 2 Associate the policy with an EPG that will consume it by sending a post with XML such as the following example:

Example:

```
<fvAEPg
     prio="unspecified" prefGrMemb="exclude" pcEnfPref="unenforced"
     name="ep2" matchT="AtleastOne" isAttrBasedEPg="no" fwdCtrl=""
     dn="uni/tn-t001/ap-ap2/epg-ep2">
   <fvRsDomAtt tDn="uni/vmmp-VMware/dom-vs1" resImedcy="lazy"
     primaryEncap="unknown" netflowPref="disabled" instrImedcy="lazy" encapMode="auto"
     encap="unknown" delimiter="" classPref="encap"/>
   <fvRsCustQosPol tnQosCustomPolName="vrfQos001"/>
   <fvRsBd tnFvBDName="default"/>
</fvAEPg>
```

Troubleshooting Cisco APIC QoS Policies

The following table summarizes common troubleshooting scenarios for the Cisco APIC QoS.

Problem	Solution

Unable to update a configured QoS policy.	**1.** Invoke the following API to ensure that `qospDscpRule` is present on the leaf.
	`GET https://192.0.20.123/api/node/class/qospDscpRule.xml`
	2. Ensure that the QoS rules are accurately configured and associated to the EPG ID to which the policy is attached.
	Use the following NX-OS style CLI commands to verify the configuration.
	`leaf1#`
	`show vlan`
	`leaf1#`
	`show system internal aclqos qos policy detail`
	`apic1#`
	`show running-config tenant` *`tenant-name`* **`policy-map type qos`** *`custom-qos-policy-name`*
	`apic1#`
	`show running-config tenant` *`tenant-name`* **`application`** *`application-name`* **`epg`** *`epg-name`*