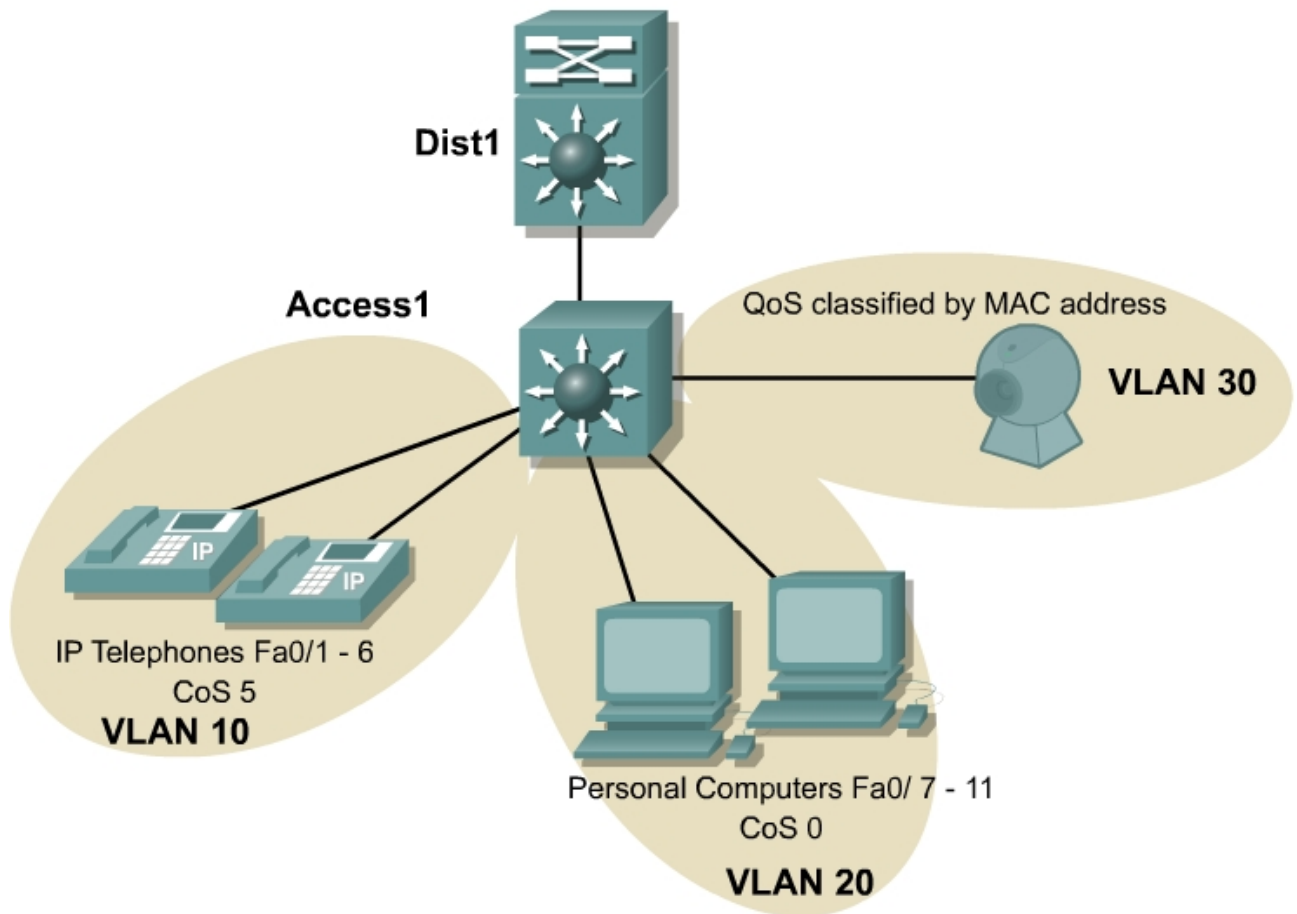


Lab 8.1.10.2 Introduction to the Modular QoS Command-Line Interface



Objective

Configuring Quality of Service (QoS) involves classifying, marking, and policing traffic flows. It is often necessary to apply the same rules to various classes of traffic or to apply the same policy to many interfaces on a switch. The IOS uses a Modular QoS Command line interface (MQC) to avoid repetition and to make it easier to modify settings.

This lab introduces the MQC, which is an important part of the QoS configuration on an IOS based switch or router.

This lab also introduces the concept of the Differentiated Services Code Point (DSCP), which is used to mark packets with a QoS identifier.

This lab is designed for use with the Catalyst 3550 switch and continues to build on Lab 8.9.1.

Scenario

The marketing department access switch has been configured for Layer 2 class of service (CoS). However, this does not provide a QoS indicator that can be carried end-to-end through the network. To achieve this, the packets must be marked at Layer 3 using the DSCP field in the IP packet as the packets move into the distribution-layer switch. In the previous lab, traffic was marked using the CoS as frames entered the access-layer switch. In this lab a Layer 3 DSCP will be set according to the existing Layer 2 CoS of the frames.

The marketing department personnel occasionally use an IP based audio-conferencing phone. Since this is not used often, a switch port does not need to be reserved on every access-layer switch. The workers would like the ability to roam and be able to unplug the nearest network device and plug the audio-conferencing phone into that port. Ensure that this device receives the same treatment as other voice traffic in the network.

Step 1

Configure the host names for the switches. Name the configured switch from the previous lab **Access1**. Name the new distribution layer 3550 switch **Dist1**.

```
Switch#configure terminal
Enter configuration commands, one per line.  End with
CNTL/Z.
Switch(config)#hostname Access1
Access1(config)#

Switch#configure terminal
Enter configuration commands, one per line.  End with
CNTL/Z.
Switch(config)#hostname Dist1
Dist1(config)#
```

Now configure a gigabit trunk between the access and distribution layer switches.

```
Access1(config)#interface gigabitethernet 0/1
Access1(config-if)#switchport mode trunk
Access1(config-if)#^Z

Dist1(config)#interface gigabitethernet 0/1
Dist1(config-if)#switchport trunk encapsulation dot1q
Dist1(config-if)#switchport mode trunk
Dist1(config-if)#exit

Access1(config)#interface gigabitEthernet 0/1
Access1(config-if)#switchport mode trunk

Dist1(config)#interface gigabitEthernet 0/1
Dist1(config-if)#switchport trunk encapsulation dot1q
Dist1(config-if)#switchport mode trunk
```

Step 2

Before using the Catalyst 3550, the QoS functionality must be enabled by using the **mls qos** command.

```
Dist1(config)#mls qos

Dist1#show mls qos
QoS is disabled

Dist1#configure terminal
Enter configuration commands, one per line.  End with
CNTL/Z.
Dist1(config)#mls qos

Dist1#show mls qos
QoS is enabled
```

This step does not apply to the Catalyst 2950 since the QoS features of the 2950 are always available.

Step 3

Traffic from the audio-conference device must be identified before it can be classified. In the previous lab, the incoming access port was used to identify frames and set the CoS. An incoming port cannot be used in this lab because the marketing people want to roam and move the device from port to port.

One mechanism that could be used to identify traffic from the audio-conference device is an IP access list. The problem with this solution is that the audio conference device will require different IP addresses if it is used on ports in different VLANs. This will make it more difficult to manage the ACL. The solution is to use a MAC-based ACL.

Note In this lab an example MAC address will be used. Substitute the MAC address of an available PC to facilitate testing.

Configure a MAC ACL on the distribution layer switch to identify traffic originating from the audio conference device.

```
Dist1(config)#mac access-list extended AUDIO-CONFERENCE
Dist1(config-ext-macl)#permit host 0000.0a00.0111 any
Dist1(config-ext-macl)#^Z

Dist1(config)#mac access-list extended AUDIO-CONFERENCE
Dist1(config-ext-macl)#permit host 0008.74c7.9648 any
```

Step 4

Verify the configuration of the MAC ACL using the **show access-lists** command.

```
Dist1#show access-lists
Extended MAC access list AUDIO-CONFERENCE
```

```
permit host 0000.0a00.0111 any

Dist1#sh access-lists
Extended MAC access list AUDIO-CONFERENCE
permit host 0008.74c7.9648 any
```

Step 5

The first component of the Modular QoS CLI is the **class-map**. The **class-map** defines the traffic types that will receive the same QoS treatment.

The **class-map** command uses various match statements to define the traffic. If **match-all** is used, the traffic must satisfy all of the match statements. If **match-any** is used, traffic that matches any of the statements will join the traffic class.

Each class-map is given a name that is used to reference the class-map.

Create a class-map called **VOICE-TRAFFIC** that matches all of the criteria specified.

```
Dist1(config)#class-map match-all VOICE-TRAFFIC
```

The **match** command is used to identify traffic that will become part of the class-map. Use the following command to examine the possible criteria for a match.

```
Dist1(config-cmap)#match ?
```

Create a match using the named ACL that was previously defined.

```
Dist1(config-cmap)#match access-group name AUDIO-CONFERENCE
Dist1(config-cmap)#^Z

Dist1(config)#class-map match-all VOICE-TRAFFIC
Dist1(config-cmap)#match access-group name AUDIO-CONFERENCE
```

Step 6

Verify the configuration using the **show class-map** command.

```
Dist1#show class-map
Class Map match-any class-default (id 0)
Match any
Class Map match-all VOICE-TRAFFIC (id 2)
Match access-group name AUDIO-CONFERENCE

Dist1#
```

The switch will automatically create a class-map called **class-default**. Match statements can also be assigned to this class-map.

Step 7

After defining the traffic class with the class-map statement, define the actions that should be taken on each class of traffic with the **policy-map** statement. Like the class-map, the policy-map is given a name. In this lab, the policy map will be called **FROM-ACCESS-LAYER**.

```
Dist1(config)#policy-map FROM-ACCESS-LAYER
```

The format of the policy-map is a reference to a traffic class and one or more actions that must be applied to the traffic. For the traffic class named **VOICE-TRAFFIC**, specify that the DSCP should be set to 40. When the **set** command is configured, use the question mark (?) to examine the other actions that can be taken on a traffic class.

```
Dist1(config-pmap)#class VOICE-TRAFFIC
Dist1(config-pmap-c)#set ip dscp 40
```

After specifying an action for traffic originating from the audio-conference device, determine the QoS requirements of traffic originating from any other hosts attached to the access-layer switch. Assume that suitable CoS values have been provided by the access-layer switch and configure the class-default policy so that the CoS value of all other traffic is trusted.

```
Dist1(config-pmap)#class class-default
Dist1(config-pmap-c)#trust cos
Dist1(config-pmap-c)#^Z

Dist1(config)#policy-map FROM-ACCESS-LAYER
Dist1(config-pmap)#class VOICE-TRAFFIC
Dist1(config-pmap-c)#set ip dscp 40
Dist1(config-pmap-c)#class class-default
Dist1(config-pmap-c)#trust cos
```

Step 8

Use the **show policy-map** command to verify the policy-map.

```
Dist1#show policy-map
Policy Map FROM-ACCESS-LAYER
  class VOICE-TRAFFIC
    set ip dscp 40
  class class-default
    trust cos
```

Step 9

The final configuration step for MCQ is applying the policy to an interface. This is accomplished by using the **service-policy** command on the required interface.

```
Dist1(config)#interface gigabitEthernet 0/1
Dist1(config-if)#service-policy input FROM-ACCESS-LAYER
Dist1(config-if)#^Z

Dist1(config)#interface gigabitEthernet 0/1
Dist1(config-if)#service-policy input FROM-ACCESS-LAYER
```

Step 10

Use the **show mls qos interface gigabitEthernet 0/1** command to verify that the service-policy has been applied to the interface correctly.

```
Dist1#show mls qos interface gigabitEthernet 0/1
GigabitEthernet0/1
Attached policy-map for Ingress: FROM-ACCESS-LAYER
trust state: not trusted
trust mode: not trusted
COS override: dis
default COS: 0
DSCP Mutation Map: Default DSCP Mutation Map
trust device: none

Dist1#
```