Chapter 2

Class of Service Using IPv6 DiffServ

Now that the Internet and other router-based networks carry a variety of traffic, such as voice, video, and best-effort data, customers have asked service providers to guarantee the delivery of certain class-of-service (CoS) parameters for different traffic classes. CoS assigns different levels of service, such as guaranteed bandwidth or minimal delay, to different traffic classes. This customer need has been a challenge to service providers, because the exact number of CoS parameters and the number of categories available for different traffic flows (the *granularity* of the CoS offering) have differed not only from network to network, but often from one section of the same network to another. To be of any use to customers, consistent CoS performance must be provided end-to-end.

Service providers seeking a standard implementation for CoS often use Differentiated Services (DiffServ) to provide a consistent CoS method for varied networks employing IP version 4 (IPv4), IP version 6 (IPv6), and Multiprotocol Label Switching (MPLS) label-switched paths (LSPs). DiffServ is defined in a series of RFCs and Internet drafts, especially RFC 2474, *Definition of the Differ entiated Services Field (DS Field) in the IPv4 and IPv6 Header s*, and RFC 2475, *An Architectur e for Differentiated Services* . These form the basis for the Juniper Networks implementation of DiffServ. This chapter shows how to implement end-to-end DiffServ on M-series routers for IPv6.

This chapter assumes the reader is familiar with CoS operation on Juniper Networks routers. For more information on CoS configuration, see the *JUNOS Netw ork Interfaces and Class of Service Configur ation Guide* .

This feature guide covers these topics:

Overview

CoS is the assignment of traffic flows to different service levels. Service providers can use router-based CoS features to define service levels that provide different delay, jitter (delay variation), and packet loss characteristics to particular applications served by specific traffic flows.

Usually, IP routers forward packets independently and without any control on throughput or delay. This is known as *best-effort* service. This service is as good as the network equipment and links, and the result is satisfactory for many traditional IP applications emphasizing data delivery, such as e-mail or Web browsing. However, newer IP applications such as real-time video and audio (or voice) require lower delay, jitter, and loss parameters than simple best-effort networks can provide. CoS is intended for networks supporting these types of time-sensitive video and audio applications.

A router cannot compromise best-effort forwarding performance in order to deliver CoS features, because this merely trades one problem for another. When CoS features are enabled, they must allow routers to better process critical packets as well as best-effort traffic flows, even during times of congestion. Network throughput is determined by a combination of available bandwidth and delay. CoS guarantees a minimum bandwidth dedicated to a service class.

The main impact of CoS on network delay is in queueing delays, when packets are normally queued for output in the order of arrival, regardless of service class. Queueing delays increase with network congestion and often result in lost packets when queue buffers overflow. The other two elements of overall network delay, serial transmission delays determined by link speeds and propagation delays determined by media type, are not affected by CoS settings.

Any CoS implementation must work consistently end-to-end through the network. A standards-based, vendor-neutral CoS implementation satisfies this requirement best. Juniper Networks CoS features interoperate with other vendors' CoS implementations because they are based on IETF DiffServ standards.

DiffServ specifications establish a six-bit field in the IPv4 and IPv6 packet header to indicate the service class that should be applied to the packet. The bit values in the DiffServ field form DiffServ code points (DSCPs) that can be set by the application or a router on the edge of a DiffServ-enabled network.

Table 5 shows the mapping of DiffServ service class meanings (aliases) to DSCPs.

Table 5: Default DSCP Mappings

DiffServ Service Class Alias	IPv4 and IPv6 DSCP Mapping
ef	101110
af11	001010
af12	001100
af13	001110
af21	010010
af22	010100
af23	010110
af31	011010
af32	011100
af33	011110
af41	100010
af42	100100
af43	100110
be	000000
cs1	001000
cs2	010000
cs3	011000
cs4	100000
cs5	101000
nc1/cs6	110000
nc2/cs7	111000

None of the aliases are established by DiffServ specifications. The aliases are well-known only through usage. For example, it is widely accepted that the alias for DSCP 101110 is ef (expedited forwarding). The 21 well-known DSCPs establish 5 DiffServ service classes:

Best-effort (be)—The router does not apply any special CoS handling to packets with 000000 in the DiffServ field, a backward compatibility feature. There is usually a high probability that these packets will be dropped under congested network conditions.

Assured forwarding (af)—The router offers a high level of assurance that the packets are delivered as long as the packet flow from the customer stays within a certain service profile (the service provider defines the values). The router accepts excess traffic, but applies a random early discard (RED) drop profile to decide if the excess packets should be dropped and not forwarded. Three drop probabilities (low, medium, and high) are defined for this service class.

Expedited forwarding (ef)—The router delivers assured bandwidth, low loss, low delay, and low delay variation (jitter) end-to-end for packets in this service class. Routers accept excess traffic in this class, but in contrast to assured forwarding, out-of-profile expedited-forwarding packets can be forwarded out of sequence or dropped.

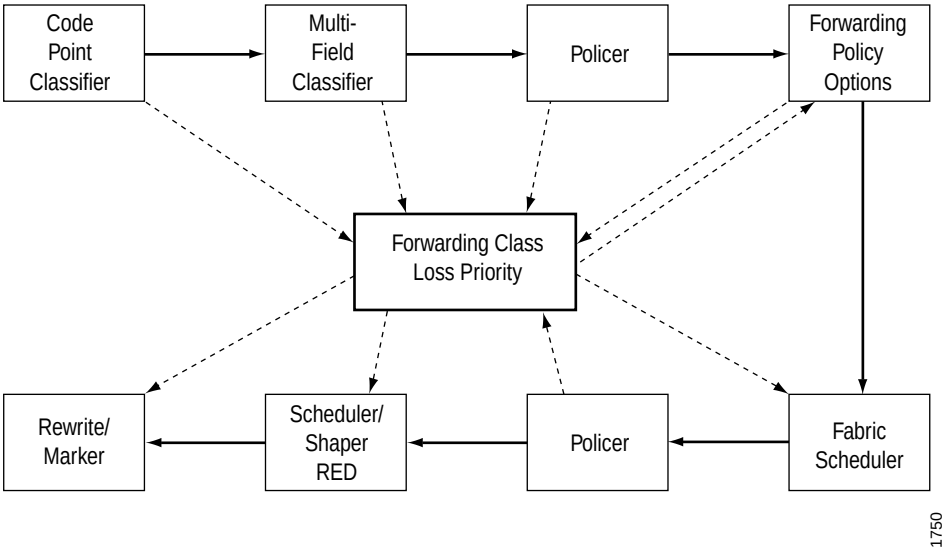
Conversational services (cs)—The router delivers assured (usually low) bandwidth with low delay and jitter for packets in this service class. Packets can be dropped, but never delivered out of sequence. Packetized voice is a good example of a conversational service.

Network control (nc)—The router delivers packets in this service class with a low priority (these packets are not delay-sensitive). Typically, these packets represent routing protocol hello or keepalive messages and loss of these packets jeopardizes proper network operation, so delay is preferable to discard.

Although CoS methods such as DiffServ specify the position and length of the DSCP in the packet header, the implementation of the router mechanisms to deliver DiffServ internally is vendor-specific. CoS functions in JUNOS software are configured through a series of mechanisms that you can configure individually or in combination to define particular service offerings.

Figure 8 shows the components of the JUNOS CoS features, illustrating the sequence in which they interact.

Figure 8: Packet Flow Through CoS-Configurable Components



You can configure one or more of the following JUNOS CoS mechanisms:

- Classifiers—Allow you to associate incoming packets with a forwarding class and packet loss priority (PLP). Two general types of classifiers are supported:
  - Behavior aggregate (BA) or code point traffic classifiers allow you to set the forwarding class and PLP based on DSCP.
  - Multifield (MF) traffic classifiers allow you to set the forwarding class and PLP based on firewall filter rules. This is usually done at the edge of the network for packets that do not have valid DSCPs in the packet headers.
- Forwarding classes—Allow you to set the scheduling and marking of packets as they transit the router. Known as ordered aggregates in the DiffServ architecture, the forwarding class plus the loss priority determine the router's per-hop behavior (PHB in DiffServ) for CoS.
- Loss priorities—Allow you to set the priority of dropping a packet before it is sent. Loss priority affects the scheduling of a packet without affecting the packet's relative ordering.
- Forwarding policy options—Allow you to associate forwarding classes with next hops. Forwarding policy also allows you to create classification overrides, which assign forwarding classes to sets of prefixes
- Transmission scheduling and rate control—Provide you with a variety of tools to manage traffic flows:
  - Schedulers—Allow you to define the priority, bandwidth, delay buffer size, rate control status, and RED drop profiles to be applied to a particular forwarding class for packet transmission.
  - Fabric schedulers—For T-series and M320 routing platforms only, fabric schedulers allow you to identify a packet as high or low priority based on its forwarding class, and to associate schedulers with the fabric priorities.
  - Policers for traffic classes—Allow you to limit traffic of a certain class to a specified bandwidth and burst size. Packets exceeding the policer limits can be discarded, or can be assigned to a different forwarding class or to a different loss priority, or to both. You define policers with filters that can be associated with input or output interfaces.
- Rewrite markers—Allow you to redefine the DSCP value of outgoing packets. Rewriting or marking outbound packets is useful when the routing platform is at the border of a network and must alter the code points to meet the policies of the targeted peer.

M-series routers have only four queues built into the hardware. T-series and M320 routing platforms can be configured for up to eight queues. If a classifier does not assign a packet to any other queue (for example, for other than well-known DSCPs that have not been added to the classifier), the packet is assigned by default to the class associated with queue 0 (Q0).

Table 6 shows the four forwarding classes and queues that Juniper Networks classifiers assign a packet based on the DSCP values in arriving packet headers.

Table 6: Default Forwarding Classes

Forwarding Class Name	Queue
best-effort	queue 0
expedited-forwarding	queue 1
assured-forwarding	queue 2
network-control	queue 3

Each forwarding class has an associated scheduler priority. Only two forwarding classes, best-effort and network-control (Q0 and Q3), are actually referenced in the default scheduler configuration. However, you can manually configure resources for the expedited-forwarding and assured-forwarding classes (Q1 and Q2).

The default scheduler settings are not visible in the output of the show class-of-service command; rather, they are implicit.

Default scheduler

```
[edit class-of-service]
schedulers {
    network-control {
        transmit-rate percent 5;
        buffer-size percent 5;
        priority low;
        drop-profile-map loss-priority any protocol any;
        drop-profile terminal;
    }
    best-effort {
        transmit-rate percent 95;
        buffer-size percent 95;
        priority low;
        drop-profile-map loss-priority any protocol any;
        drop-profile terminal;
    }
}
drop-profiles {
    terminal {
        fill-level 100 drop-probability 100;
    }
}
```

By default, the best-effort forwarding class (Q0) receives 95 percent of the output link bandwidth and buffer space, and the network-control forwarding class (Q3) receives 5 percent of the output link bandwidth and buffer space. The default drop profile provides *tail drop*, where the buffer fills and then discards all packets until there is space in the buffer again. There are no schedulers for the expedited-forwarding or assured-forwarding classes because by default no resources are assigned to Q1 and Q2.

Table 7 shows the default forwarding class and packet loss priority (PLP) for the well-known DSCPs. It is important to note that although several DSCPs map to the expedited-forwarding and assured-forwarding classes, by default no resources are assigned to these forwarding classes. All of these settings can be changed through configuration.

Table 7: Default Behavior Aggregate Classification

DSCP and DSCP IPv6	Forwarding Class	PLP
ef	expedited-forwarding	low
af11	assured-forwarding	low
af12	assured forwarding	high
af13	assured forwarding	high
af21	best-effort	low
af22	best-effort	low
af23	best-effort	low
af31	best-effort	low
af32	best-effort	low
af33	best-effort	low
af41	best-effort	low
af42	best-effort	low
af43	best-effort	low
be	best-effort	low
cs1	best-effort	low
cs2	best-effort	low
cs3	best-effort	low
cs4	best-effort	low
cs5	best-effort	low
nc1/cs6	network-control	low
nc2/cs7	network control	low
other	best-effort	low

All af classes other than af1*x* are mapped to best-effort, since RFC 2597 prohibits a node from aggregating classes. In effect, mapping to best-effort implies that the node does not support that class.

Typically, rewrites of the DSCPs on outgoing packets are done once, when packets enter the DiffServ portion of the network, either because the packets do not arrive from the customer with the proper DSCP bit set or because the service provider wishes to verify the that customer has set the DSCP properly. CoS schemes that accept the DSCP and classify and schedule traffic solely on DSCP value perform behavior aggregate (BA) DiffServ functions and do not usually rewrite the DSCP. DSCP rewrites typically occur in multifield (MF) DiffServ scenarios.

Table 8 shows the router forwarding classes associated with each well-known DSCP code point and the resources assigned to their output queues.

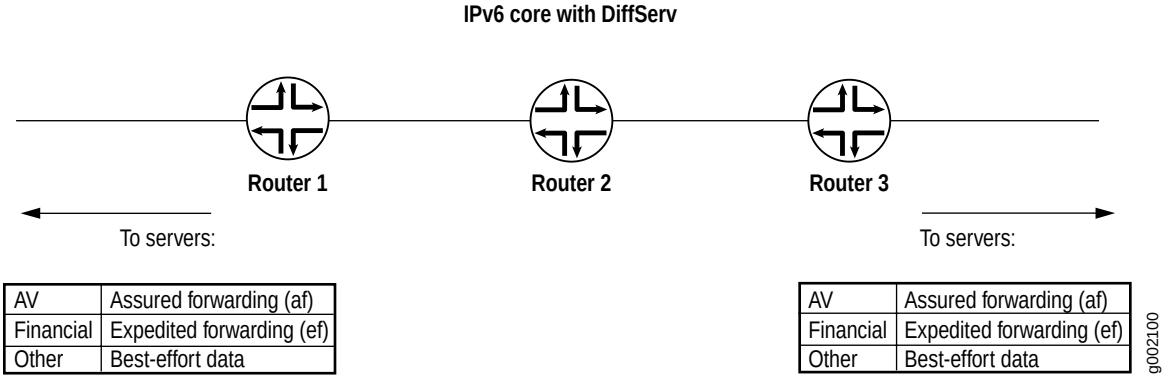
Table 8: Classification Forwarding Classes and Queues

DCSP Alias	DSCP Bits	Forwarding Class	PLP	Queue
ef	101110	expedited-forwarding	low	1
af11	001010	assured-forwarding	low	2
af12	001100	assured-forwarding	high	2
af13	001110	assured-forwarding	high	2
af21	010010	best-effort	low	0
af22	010100	best-effort	low	0
af23	010110	best-effort	low	0
af31	011010	best-effort	low	0
af32	011100	best-effort	low	0
af33	011110	best-effort	low	0
af41	100010	best-effort	low	0
af42	100100	best-effort	low	0
af43	100110	best-effort	low	0
be	000000	best-effort	low	0
cs1	001000	best-effort	low	0
cs2	010000	best-effort	low	0
cs3	011000	best-effort	low	0
cs4	100000	best-effort	low	0
cs5	101000	best-effort	low	0
nc1/cs6	110000	network-control	low	3
nc2/cs7	111000	network-control	low	3
other	—	best-effort	low	0

The example in this chapter essentially assigns expedited forwarding to Q1 and a subset of the assured forwarding classes (af1*x*) to Q2, and distributes resources among all four forwarding classes.

Figure 9 shows the topology of the three routers and links that are used as a case study in this chapter.

Figure 9: Basic IPv6 DiffServ Topology



In this case study, the service provider has agreed to provide high-priority delivery of packets for two applications between the customer's servers at two sites. The first application generates streams of high-definition audio-visual (television) packet flows and the second generates large quantities of time-sensitive financial information. In all cases, the packet flow is from server to server. The service provider marks the packets appropriately as they enter the network from either site, configures special queues and forwarding classes for this traffic on the three routers, and uses DiffServ for IPv6 for this purpose.

Routers 1 and 3 use MF classifiers on the customer-facing interfaces to detect high-priority packets and rewrite the DSCPs appropriately. Best-effort data and network control packets are not affected. All three routers are configured with consistent schedulers and resources to handle high-priority packets properly.

Table 9 shows the resources assigned to the four forwarding classes in this example.

Table 9: Resources Assigned to Queues

Queue	Forwarding Class	Transmit Rate	Buffer Size	Priority
0	be (data)	40%	40%	low
1	ef (financial)	10%	10%	high
2	af (audio-visual)	45%	45%	high (with RED)
3	nc (network control)	5%	5%	low

The table shows how the 95 percent of output link transmission rate and buffer size (queue) resources assigned by default to Q0 (best-effort) are distributed to Q1 (expedited forwarding) and Q2 (assured forwarding). The audio-visual traffic consumes more bandwidth than other applications, but the financial information, although critical, is carried in fewer packets. In keeping with DiffServ specifications, a RED drop profile is applied to the assured forwarding class. The financial data has a strict set of traffic parameters that must be respected.

The three DiffServ assured forwarding classes supported (af11, af12, and af13, with low, medium, and high packet drop probability, respectively) are distinguished by using a low PLP and RED drop profile for af11 and a high PLP and RED for af12 and af13. All of these parameters should be closely monitored initially for performance and adjusted as necessary.

System Requirements

To implement CoS with DiffServ for IPv6, your system must meet these minimum requirements:

JUNOS Release 6.3 or later

Three Juniper Networks M-series or T-series routing platforms

For M-series routers, Enhanced FPCs capable of supporting DSCPs and, for MF classifiers, Internet Processor II ASICs

Terms and Acronyms

classifier—A method of reading a sequence of bits in a packet header or label and determining the packet's forwarding class.

class of service—A set of forwarding class parameters that define different treatment for different traffic flows.

Differentiated Services (DiffServ)—A standards-based method of associating CoS parameters with traffic flows and their forwarding classes.

Differentiated Services code point (DSCP)—Values for a 6-bit field defined for IPv4 and IPv6 packet headers that can be used to enforce CoS distinctions in routers.

Configuring CoS with IPv6 DiffServ

You override the default packet forwarding behavior of the router when you configure CoS using IPv6 DiffServ. The three main areas of configuration are classifiers (by default, the router does not classify packets using DiffServ), scheduling (by default, the router only has two queues enabled), and rewrite rules (by default, the router does not rewrite CoS bits).

To implement CoS using IPv6 DiffServ, you must configure the following:

Configuring a Firewall Filter for an MF Classifier on Customer Interfaces on page 88

Applying the Firewall Filter to Customer Interfaces on page 89

Assigning Forwarding Classes to Output Queues on page 89

Configuring Rewrite Rules on page 90

Applying Rewrite Rules to an Interface on page 90

Configuring BA Classifiers on page 91

Applying a BA Classifier to an Interface on page 91

Configuring RED Drop Profiles on page 92

Configuring Schedulers on page 92

Configuring Scheduler Maps on page 93

Applying a Scheduler Map to an Interface on page 93

To apply your knowledge, visit these sections:

Example: CoS with IPv6 DiffServ Configuration on page 94

Checking Your Work on page 104

Configuring a Firewall Filter for an MF Classifier on Customer Interfaces

You configure an MF classifier for IPv6 to detect packets of interest to CoS and assign the packet to the proper forwarding class independently of DSCP. To configure an MF classifier on a customer-facing link, configure a policer for the expedited forwarding traffic and a firewall filter to classify traffic.

```
[edit firewall]
policer ef-FIN-Policer-Profile {
    if-exceeding {
      bandwidth-percent 10;
      burst-size-limit 2k;
    }
    then loss-priority high;
}
family inet6 {
    filter mf-classifier {
      filter-specific;
      term AV {
        from {
          destination-address {
            0:0:FFFF:172.16.79.11;
          }
        }
        then {
          loss-priority low;
          fowarding-class af-AV-class;
        }
      }
      term Finance {
        from {
          destination-address {
            O:0:FFFF:172.16.79.63;
          }
        }
        then {
          policer ef-FIN-Policer-Profile;
          forwarding-class ef-FIN-class;
        }
      }
      term Network-Control {
        from {
          traffic-class 192; # 192 is the 110000 traffic class.
        }
        then {
          forwarding-class nc-CONTROL-class; # This is network control traffic.
        }
      }
      term Data {
        then forwarding-class be-DATA-class; # The rest is data.
      }
    }
}
```

Applying the Firewall Filter to Customer Interfaces

You apply an MF classifier firewall filter for IPv6 to customer interfaces. To apply an MF classifier firewall filter on customer-facing links, apply the classifier as an input filter at the [edit interfaces] hierarchy level.

```
[edit interfaces]
so-0/0/1 {
    unit 0 {
        family inet {
            address 192.168.54.1/24;
        }
        family inet6 {
            filter {
                input mf-classifier;
            }
            address 0:0:FFFF:192.168.54.1/120;
        }
    }
}
```

Assigning Forwarding Classes to Output Queues

You must assign the forwarding classes established by the MF classifier to output queues. To assign a forwarding class to an output queue, include the forwarding-classes statement at the [edit class-of-service] hierarchy level.

```
[edit class-of-service]
forwarding-classes {
    queue 0 be-DATA-class;
    queue 1 ef-FIN-class;
    queue 2 af-AV-class;
    queue 3 nc-CONTROL-class;
}
```

Configuring Rewrite Rules

You configure rewrite rules to replace DSCPs on packets received from the customer with the values expected by other routers. Rewrite rules use the forwarding class information and PLP used internally by the router to establish the DSCP on outbound packets. To configure rewrite rules, include the rewrite-rules statement at the [edit class-of-service] hierarchy level.

```
[edit class-of-service]
rewrite-rules rewrite-IPv6-dscps {
    forwarding-class be-DATA-class {
        loss-priority low code points 000000;
        loss-priority high code points 000001;
    }
    forwarding-class ef-FIN-class {
        loss-priority low code points 101110;
        loss-priority high code points 101111;
    }
    forwarding-class af-AV-class {
        loss-priority low code points 001010;
        loss-priority high code points 001100;
    }
    forwarding-class nc-CONTROL-class {
        loss-priority low code points 110000;
        loss-priority high code points 110001;
    }
}
```

Applying Rewrite Rules to an Interface

To apply the configured rewrite rules, include the rewrite-rules statement at the [edit class-of-service interfaces] hierarchy level.

```
[edit class-of-service interfaces]
so-0/1/1 {
    unit 0 {
        rewrite-rules {
            dscp-ipv6 rewrite-IPv6-dscps;
        }
    }
}
```

Configuring BA Classifiers

You configure BA classifiers for IPv6 on network interfaces because the DSCPs have been explicitly rewritten on the edge routers. To configure a BA classifier for IPv6 DSCPs, include the dscp-ipv6 statement and give the classifier a name. Then import the default classifier and specify the forwarding class, loss priority, and code points for each established traffic class at the [edit class-of-service] hierarchy level.

```
[edit class-of-service]
classifiers {
    dscp-ipv6 IPv6-classifier {
        import default; # Uses the DSCP default map.
        forwarding-class be-DATA-class {
            loss-priority high code-points 000001;
        }
        forwarding-class ef-FIN-class {
            loss-priority high code-points 101111;
        }
        forwarding-class af-AV-class {
            loss-priority high code-points 001100;
        }
        forwarding-class nc-CONTROL-class {
            loss-priority high code-points 110001;
        }
    }
}
```

Applying a BA Classifier to an Interface

To apply the configured classifier, include the classifiers statement at the [edit class-of-service interfaces] hierarchy level.

```
[edit class-of-service interfaces]
so-0/1/1 {
    unit 0 {
        classifiers {
            dscp-ipv6 IPv6-classifier;
        }
    }
}
```

Configuring RED Drop Profiles

You configure RED drop profiles to determine the probability of DiffServ assured forwarding packets being discarded under congested conditions. To configure RED drop profiles for assured forwarding without the PLP bit set and with the PLP bit set, include the drop-profiles statement at the [edit class-of-service] hierarchy level.

```
[edit class-of-service]
drop-profiles {
    af-AV-normal {
        interpolate {
            fill-level [95 100];
            drop-probability [0 100];
        }
    }
    af-AV-with-PLP {
        interpolate {
            fill-level [60 70 80 90 95];
            drop-probability [80 90 95 97 100];
        }
    }
```

Assured forwarding traffic with the PLP bit set has a more aggressive drop probability than traffic without the PLP bit set.

Configuring Schedulers

You configure schedulers to assign resources, priorities, and drop profiles to output queues. To configure a scheduler, include the schedulers statement at the [edit class-of-service] hierarchy level.

```
[edit class-of-service]
schedulers {
    be-DATA-scheduler {
        transmit-rate percent 40;
        buffer-size percent 40;
        priority low;
    }
    ef-FIN-scheduler {
        transmit-rate percent 10;
        buffer-size percent 10;
        priority high;
    }
    af-AV-scheduler {
        transmit-rate percent 45;
        buffer-size percent 45;
        priority high;
        drop-profile-map loss-priority low protocol any drop-profile af-AV-normal;
        drop-profile-map loss-priority high protocol any drop-profile af-AV-with-PLP;
    }
    nc-CONTROL-scheduler {
        transmit-rate percent 5;
        buffer-size percent 5;
        priority low;
    }
}
```

Configuring Scheduler Maps

You configure a scheduler map to assign a forwarding class to a scheduler. To configure a scheduler map, include the scheduler-maps statement and scheduler name at the [edit class-of-service] hierarchy level.

```
[edit class-of-service]
scheduler-maps {
    diffserv-cos-map {
        fowarding-class be-DATA-class scheduler be-DATA-scheduler;
        fowarding-class ef-FIN-class scheduler ef-FIN-scheduler;
        fowarding-class af-AV-class scheduler af-AV-scheduler;
        fowarding-class nc-CONTROL-class scheduler nc-CONTROL-scheduler;
    }
}
```

Applying a Scheduler Map to an Interface

To apply the configured scheduler map, include the scheduler-map statement at the [edit class-of-service] hierarchy level.

```
[edit class-of-service]
interfaces {
    so-1/0/1 {
        scheduler-map diffserv-cos-map;
    }
}
```

Example: CoS with IPv6 DiffServ Configuration

Figure 10: IPv6 DiffServ Configuration

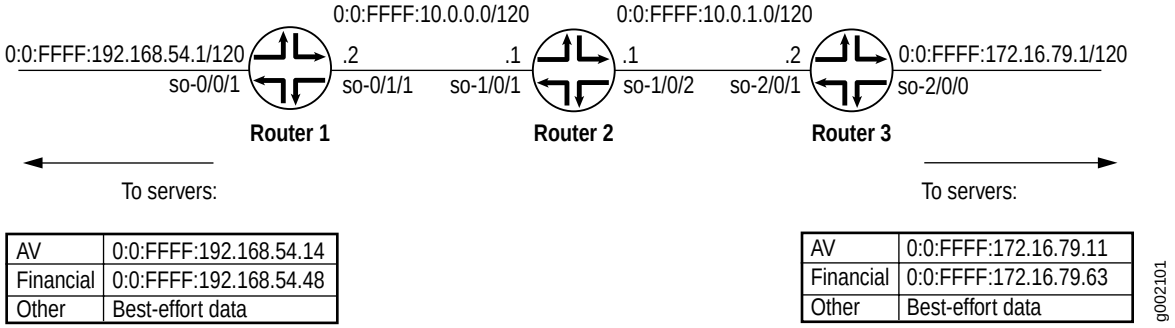


Figure 10 shows the complete topology for IPv6 DiffServ, complete with interfaces and IPv6 addresses. The IPv4-mapped IPv6 address format described in RFC 1884 is used.

Begin your configuration on Router 2, the core router. This ensures that when DiffServ is enabled on the edge routers, CoS is enabled end-to-end through the network. The core router configuration is a little simpler because no MF classification is configured in the core.

Router 2

```
[edit]
class-of-service {
    classifiers {           # Router 2 classifiers.
      dscp-ipv6 IPv6-classifier {
        import default; # Uses the DSCP default map.
        forwarding-class be-DATA-class {
          loss-priority high code-points 000001;
        }
        forwarding-class ef-FIN-class {
          loss-priority high code-points 101111;
        }
        forwarding-class af-AV-class {
          loss-priority high code-points 001100;
        }
        forwarding-class nc-CONTROL-class {
          loss-priority high code-points 110001;
        }
      }
    }
```

```
drop-profiles {              # Router 2 drop profiles.
   af-AV-normal {
      interpolate {
         fill-level [95 100];
         drop-probability [0 100];
      }
   }
   af-AV-with-PLP {
      interpolate {
         fill-level [60 70 80 90 95];
         drop-probability [80 90 95 97 100];
      }
   }
}
fowarding-classes {        # Router 2 forwarding classes.
   queue 0 be-DATA-class;
   queue 1 ef-FIN-class;
   queue 2 af-AV-class;
   queue 3 nc-CONTROL-class;
}
interfaces {                 # Router 2 class-of-service interfaces.
   so-1/0/1 {              # Connected to R1.
      scheduler-map diffserv-cos-map;
      unit 0 {
         classifiers {
            dscp-ipv6 IPv6-classifier;
         }
         rewrite-rules {
            dscp-ipv6 rewrite-IPv6-dscp;
         }
      }
   }
   so-1/0/2 {              # Connected to R3.
      scheduler-map diffserv-cos-map;
      unit 0 {
         classifiers {
            dscp-ipv6 IPv6-classifier;
         }
         rewrite-rules {
            dscp-ipv6 rewrite-IPv6-dscp;
         }
      }
   }
}
```

```
        rewrite-rules rewrite-IPv6-dscps {        # Router 2 rewrite rules.
          forwarding-class be-DATA-class {
            loss-priority low code points 000000;
            loss-priority high code points 000001;
          }
          forwarding-class ef-FIN-class {
            loss-priority low code points 101110;
            loss-priority high code points 101111;
          }
          forwarding-class af-AV-class {
            loss-priority low code points 001010;
            loss-priority high code points 001100;
          }
          forwarding-class nc-CONTROL-class {
            loss-priority low code points 110000;
            loss-priority high code points 110001;
          }
        }
        scheduler-maps {                          # Router 2 scheduler maps.
          diffserv-cos-map {
            forwarding-class be-DATA-class scheduler be-DATA-scheduler;
            forwarding-class ef-FIN-class scheduler ef-FIN-scheduler;
            forwarding-class af-AV-class scheduler af-AV-scheduler;
            forwarding-class nc-CONTROL-class scheduler nc-CONTROL-scheduler;
          }
        }
        schedulers {                              # Router 2 schedulers.
          be-DATA-scheduler {
            transmit-rate percent 40;
            buffer-size percent 40;
            priority low;
          }
          ef-FIN-scheduler {
            transmit-rate percent 10;
            buffer-size percent 10;
            priority high;
          }
          af-AV-scheduler {
            transmit-rate percent 45;
            buffer-size percent 45;
            priority high;
            drop-profile-map loss-priority low protocol any drop-profile af-AV-normal;
            drop-profile-map loss-priority high protocol any drop-profile af-AV-with-PLP;
          }
          nc-CONTROL-scheduler {
            transmit-rate percent 5;
            buffer-size percent 5;
            priority low;
          }
        }
    }
```

```
interfaces {                    # R2 interfaces.
    so-1/0/1 {                  # Connected to R1.
        unit 0 {
            family inet {
                address 10.0.0.1/24;
            }
            family inet6 {
                address 0:0:FFFF:10.0.0.1/120;
            }
        }
    }
    so-1/0/2 {                  # Connected to R3.
        unit 0 {
            family inet {
                address 10.0.1.1/24;
            }
            family inet6 {
                address 0:0:FFFF:10.0.1.1/120;
            }
        }
    }
}
```

Continue your configuration on Router 1 and Router 3, the edge routers. These routers get firewall-filter-based MF classifiers and rewrite rules for markers as well as schedulers and drop profiles on the core-facing interfaces.

Router 1

```
[edit]
class-of-service {
    classifiers {                   # Router 1 classifiers.
        dscp-ipv6 IPv6-classifier {
            import default;         # Uses the DSCP default map.
            forwarding-class be-DATA-class {
                loss-priority high code-points 000001;
            }
            forwarding-class ef-FIN-class {
                loss-priority high code-points 101111;
            }
            forwarding-class af-AV-class {
                loss-priority high code-points 001100;
            }
            forwarding-class nc-CONTROL-class {
                loss-priority high code-points 110001;
            }
        }
    }
```

```
drop-profiles {                 # Router 1 drop profiles.
   af-AV-normal {
      interpolate {
         fill-level [95 100];
         drop-probability [0 100];
      }
   }
   af-AV-with-PLP {
      interpolate {
         fill-level [60 70 80 90 95];
         drop-probability [80 90 95 97 100];
      }
   }
}
forwarding-classes {          # Router 1 forwarding classes.
   queue 0 be-DATA-class;
   queue 1 ef-FIN-class;
   queue 2 af-AV-class;
   queue 3 nc-CONTROL-class;
}
interfaces {                     # Router 1 class-of-service interfaces.
   so-0/1/1 {                 # To servers.
      scheduler-map diffserv-cos-map;
      unit 0 {
         classifiers {
            dscp-ipv6 IPv6-classifier;
         }
         rewrite-rules {
            dscp-ipv6 rewrite-IPv6-dscp;
         }
      }
   }
rewrite-rules rewrite-IPv6-dscps {              # Router 1 rewrite rules.
   forwarding-class be-DATA-class {
      loss-priority low code points 000000;
      loss-priority high code points 000001;
   }
   forwarding-class ef-FIN-class {
      loss-priority low code points 101110;
      loss-priority high code points 101111;
   }
   forwarding-class af-AV-class {
      loss-priority low code points 001010;
      loss-priority high code points 001100;
   }
   forwarding-class nc-CONTROL-class {
      loss-priority low code points 110000;
      loss-priority high code points 110001;
   }
}
```

```
    scheduler-maps {          # Router 1 scheduler map.
      diffserv-cos-map {
        forwarding-class be-DATA-class scheduler be-DATA-scheduler;
        forwarding-class ef-FIN-class scheduler ef-FIN-scheduler;
        forwarding-class af-AV-class scheduler af-AV-scheduler;
        forwarding-class nc-CONTROL-class scheduler nc-CONTROL-scheduler;
      }
    }
    schedulers {              # Router 1 schedulers.
      be-DATA-scheduler {
        transmit-rate percent 40;
        buffer-size percent 40;
        priority low;
      }
      ef-FIN-scheduler {
        transmit-rate percent 10;
        buffer-size percent 10;
        priority high;
      }
      af-AV-scheduler {
        transmit-rate percent 45;
        buffer-size percent 45;
        priority high;
        drop-profile-map loss-priority low protocol any drop-profile af-AV-normal;
        drop-profile-map loss-priority high protocol any drop-profile af-AV-with-PLP;
      }
      nc-CONTROL-scheduler {
        transmit-rate percent 5;
        buffer-size percent 5;
        priority low;
      }
    }
}
firewall {                    # Router 1 firewall policer and filter.
    policer ef-FIN-Policer-Profile {
      if-exceeding {
        bandwidth-percent 10;
        burst-size-limit 2k;
      }
      then loss-priority high;
    }
    family inet6 {
      filter mf-classifier {
        filter-specific;
        term AV {
          from {
            destination-address {
              0:0:FFFF:172.16.79.11;
            }
          }
          then {
            loss-priority low;
            forwarding-class af-AV-class;
          }
        }
```

```
            term Finance {
               from {
                  destination-address {
                     O:0:FFFF:172.16.79.63;
                  }
               }
               then {
                  policer ef-FIN-Policer-Profile;
                  forwarding-class ef-FIN-class;
               }
            }
            term Network-Control {
               from {
                  traffic-class 192; # 192 is the 110000 traffic class.
               }
               then {
                  forwarding-class nc-CONTROL-class; # This is network control traffic.
               }
            }
            term Data {
               then forwarding-class be-DATA-class; # The rest is data.
            }
         }
      }
}
interfaces {                    # Router 1 interfaces.
      so-0/0/1 {                # To servers.
         unit 0 {
            family inet {
               address 192.168.54.1/24;
            }
            family inet6 {
               filter {
                  input mf-classifier;
               }
               address 0:0:FFFF:192.168.54.1/120;
            }
         }
      }
      so-0/1/1 {                # Connected to R2.
         unit 0 {
            family inet {
               address 10.0.0.2/24;
            }
            family inet6 {
               address 0:0:FFFF:10.0.0.2/120;
            }
         }
      }
}
```

Router 3

```
[edit]
class-of-service {
    classifiers {                # Router 3 classifiers.
      dscp-ipv6 IPv6-classifier {
        import default;      # Uses the DSCP default map.
        forwarding-class be-DATA-class {
          loss-priority high code-points 000001;
        }
        forwarding-class ef-FIN-class {
          loss-priority high code-points 101111;
        }
        forwarding-class af-AV-class {
          loss-priority high code-points 001100;
        }
        forwarding-class nc-CONTROL-class {
          loss-priority high code-points 110001;
        }
      }
    }
    drop-profiles {              # Router 3 drop profiles.
      af-AV-normal {
        interpolate {
          fill-level [95 100];
          drop-probability [0 100];
        }
      }
      af-AV-with-PLP {
        interpolate {
          fill-level [60 70 80 90 95];
          drop-probability [80 90 95 97 100];
        }
      }
    }
    forwarding-classes {    # Router 3 forwarding classes.
      queue 0 be-DATA-class;
      queue 1 ef-FIN-class;
      queue 2 af-AV-class;
      queue 3 nc-CONTROL-class;
    }
    interfaces {             # Router 3 class-of-service interfaces.
      so-2/0/1 {          # To servers.
        scheduler-map diffserv-cos-map;
        unit 0 {
          classifiers {
            dscp-ipv6 IPv6-classifier;
          }
          rewrite-rules {
            dscp-ipv6 rewrite-IPv6-dscp;
          }
        }
      }
```

```
rewrite-rules rewrite-IPv6-dscps {              # Router 3 rewrite rules.
   forwarding-class be-DATA-class {
      loss-priority low code points 000000;
      loss-priority high code points 000001;
   }
   forwarding-class ef-FIN-class {
      loss-priority low code points 101110;
      loss-priority high code points 101111;
   }
   forwarding-class af-AV-class {
      loss-priority low code points 001010;
      loss-priority high code points 001100;
   }
   forwarding-class nc-CONTROL-class {
      loss-priority low code points 110000;
      loss-priority high code points 110001;
   }
}
scheduler-maps {                                # Router 3 scheduler map.
   diffserv-cos-map {
      forwarding-class be-DATA-class scheduler be-DATA-scheduler;
      forwarding-class ef-FIN-class scheduler ef-FIN-scheduler;
      forwarding-class af-AV-class scheduler af-AV-scheduler;
      forwarding-class nc-CONTROL-class scheduler nc-CONTROL-scheduler;
   }
}
schedulers {                                    # Router 3 schedulers.
   be-DATA-scheduler {
      transmit-rate percent 40;
      buffer-size percent 40;
      priority low;
   }
   ef-FIN-scheduler {
      transmit-rate percent 10;
      buffer-size percent 10;
      priority high;
   }
   af-AV-scheduler {
      transmit-rate percent 45;
      buffer-size percent 45;
      priority high;
      drop-profile-map loss-priority low protocol any drop-profile af-AV-normal;
      drop-profile-map loss-priority high protocol any drop-profile af-AV-with-PLP;
   }
   nc-CONTROL-scheduler {
      transmit-rate percent 5;
      buffer-size percent 5;
      priority low;
   }
}
```

```
firewall {                          # Router 3 firewall policer and filter.
    policer ef-FIN-Policer-Profile {
      if-exceeding {
        bandwidth-percent 10;
        burst-size-limit 2k;
      }
      then loss-priority high;
    }
    family inet6 {
      filter mf-classifier {
        filter-specific;
        term AV {
          from {
            destination-address {
              0:0:FFFF:172.16.79.11;
            }
          }
          then {
            loss-priority low;
            fowarding-class af-AV-class;
          }
        }
        term Finance {
          from {
            destination-address {
              O:0:FFFF:172.16.79.63;
            }
          }
          then {
            policer ef-FIN-Policer-Profile;
            fowarding-class ef-FIN-class;
          }
        }
        term Network-Control {
          from {
            traffic-class 192; # 192 is the 110000 traffic class.
          }
          then {
            forwarding-class nc-CONTROL-class; # This is network control traffic.
          }
        }
        term Data {
          then fowarding-class be-DATA-class; # The rest is data.
        }
      }
    }
}
```

```
interfaces {                    # Router 3 interfaces.
    so-2/0/0 {                  # To servers.
        unit 0 {
            family inet {
                address 1172.16.79.1/24;
            }
            family inet6 {
                filter {
                    input mf-classifier;
                }
                address 0:0:FFFF:172.16.79.1/120;
            }
        }
    }
    so-2/0/1 { # to R2
        unit 0 {
            family inet {
                address 10.0.1.2/24;
            }
            family inet6 {
                address 0:0:FFFF:10.0.1.2/120;
            }
        }
    }
}
```

Checking Your Work

To verify that your CoS using IPv6 DiffServ configuration is correct, use the following commands:

show class-of-service classifier type dscp-ipv6

show class-of-service rewrite-rule type dscp-ipv6

show class-of-service interface

show class-of-service forwarding-table classifier mapping

show class-of-service forwarding-table rewrite-rule mapping

show class-of-service scheduler-map *scheduler-map-name*

show class-of-service forwarding-table scheduler-map

The following section shows the output of these commands used with the configuration example.

DiffServ Classifiers

user@R1> **show class-of-service classifier type dscp-ipv6**

Classifier: dscp-ipv6-default, Code point type: dscp-ipv6, Index: 4

Code point	Forwarding class	Loss priority
000000	be-DATA-class	low
000001	be-DATA-class	low
000010	be-DATA-class	low
000011	be-DATA-class	low
000100	be-DATA-class	low
000101	be-DATA-class	low
000110	be-DATA-class	low
000111	be-DATA-class	low
001000	be-DATA-class	low
001001	be-DATA-class	low
001010	af-AV-class	low
001011	be-DATA-class	low
001100	af-AV-class	high
001101	be-DATA-class	low
001110	af-AV-class	high
001111	be-DATA-class	low
010000	be-DATA-class	low
010001	be-DATA-class	low
010010	be-DATA-class	low
010011	be-DATA-class	low
010100	be-DATA-class	low
010101	be-DATA-class	low
010110	be-DATA-class	low
010111	be-DATA-class	low
011000	be-DATA-class	low
011001	be-DATA-class	low
011010	be-DATA-class	low
011011	be-DATA-class	low
011100	be-DATA-class	low
011101	be-DATA-class	low
011110	be-DATA-class	low
011111	be-DATA-class	low
100000	be-DATA-class	low
100001	be-DATA-class	low
100010	be-DATA-class	low
100011	be-DATA-class	low
100100	be-DATA-class	low
100101	be-DATA-class	low
100110	be-DATA-class	low
100111	be-DATA-class	low
101000	be-DATA-class	low
101001	be-DATA-class	low
101010	be-DATA-class	low
101011	be-DATA-class	low
101100	be-DATA-class	low
101101	be-DATA-class	low
101110	ef-FIN-class	low
101111	be-DATA-class	low
110000	nc-CONTROL-class	low
110001	be-DATA-class	low
110010	be-DATA-class	low
110011	be-DATA-class	low
110100	be-DATA-class	low
110101	be-DATA-class	low
110110	be-DATA-class	low
110111	be-DATA-class	low

111000	nc-CONTROL-class	low
111001	be-DATA-class	low
111010	be-DATA-class	low
111011	be-DATA-class	low
111100	be-DATA-class	low
111101	be-DATA-class	low
111110	be-DATA-class	low
111111	be-DATA-class	low

Classifier: IPv6-classifier, Code point type: dscp-ipv6, Index: 18301

Code point	Forwarding class	Loss priority
000000	be-DATA-class	low
000001	be-DATA-class	high
000010	be-DATA-class	low
000011	be-DATA-class	low
000100	be-DATA-class	low
000101	be-DATA-class	low
000110	be-DATA-class	low
000111	be-DATA-class	low
001000	be-DATA-class	low
001001	be-DATA-class	low
001010	af-AV-class	low
001011	be-DATA-class	low
001100	af-AV-class	high
001101	be-DATA-class	low
001110	af-AV-class	high
001111	be-DATA-class	low
010000	be-DATA-class	low
010001	be-DATA-class	low
010010	be-DATA-class	low
010011	be-DATA-class	low
010100	be-DATA-class	low
010101	be-DATA-class	low
010110	be-DATA-class	low
010111	be-DATA-class	low
011000	be-DATA-class	low
011001	be-DATA-class	low
011010	be-DATA-class	low
011011	be-DATA-class	low
011100	be-DATA-class	low
011101	be-DATA-class	low
011110	be-DATA-class	low
011111	be-DATA-class	low
100000	be-DATA-class	low
100001	be-DATA-class	low
100010	be-DATA-class	low
100011	be-DATA-class	low
100100	be-DATA-class	low
100101	be-DATA-class	low
100110	be-DATA-class	low
100111	be-DATA-class	low
101000	be-DATA-class	low
101001	be-DATA-class	low
101010	be-DATA-class	low
101011	be-DATA-class	low
101100	be-DATA-class	low
101101	be-DATA-class	low
101110	ef-FIN-class	low
101111	ef-FIN-class	high
110000	nc-CONTROL-class	low
110001	nc-CONTROL-class	high
110010	be-DATA-class	low
110011	be-DATA-class	low

```
110100          be-DATA-class               low
110101          be-DATA-class               low
110110          be-DATA-class               low
110111          be-DATA-class               low
111000          nc-CONTROL-class            low
111001          be-DATA-class               low
111010          be-DATA-class               low
111011          be-DATA-class               low
111100          be-DATA-class               low
111101          be-DATA-class               low
111110          be-DATA-class               low
111111          be-DATA-class               low
```

Rewrite Rules

```
user@R1> show class-of-service rewrite-rule type dscp-ipv6
Rewrite rule: dscp-ipv6-default, Code point type: dscp-ipv6, Index: 20
  Forwarding class              Loss priority      Code point
  be-DATA-class                 low                000000
  be-DATA-class                 high               000000
  ef-FIN-class                  low                101110
  ef-FIN-class                  high               101110
  af-AV-class                   low                001010
  af-AV-class                   high               001100
  nc-CONTROL-class              low                110000
  nc-CONTROL-class              high               111000

Rewrite rule: rewrite-IPv6-dscp, Code point type: dscp-ipv6, Index: 58077
  Forwarding class              Loss priority      Code point
  be-DATA-class                 low                000000
  be-DATA-class                 high               000001
  ef-FIN-class                  low                101110
  ef-FIN-class                  high               101111
  af-AV-class                   low                001010
  af-AV-class                   high               001100
  nc-CONTROL-class              low                110000
  nc-CONTROL-class              high               110001
```

Class-of-Service Interfaces

```
user@R1> show class-of-service interface
...
Physical interface: so-0/0/1, Index: 141
Queues supported: 4, Queues in use: 4
  Scheduler map: diffserv-cos-map, Index: -543019056

  Logical interface: so-0/0/1.0, Index: 68
    Object       Name                Type              Index
    Rewrite      rewrite-IPv6-dscp   dscp-ipv6           58077
    Rewrite      exp-default         exp                    21
    Classifier   IPv6-classifier     dscp-ipv6           18301
    Classifier   exp-default         exp                     5

...
Physical interface: so-0/1/1, Index: 144
Queues supported: 4, Queues in use: 4
  Scheduler map: <default>, Index: -113795564

  Logical interface: so-0/1/1.0, Index: 69
    Object       Name                Type              Index
    Rewrite      exp-default         exp                    21
    Classifier   exp-default         exp                     5
    Classifier   ipprec-compatibility ip                     8
```

Classifier Mapping

```
user@R1> show class-of-service forwarding-table classifier mapping
                                  Table Index/
Interface     Index     Q num     Table type
so-0/0/1.0       68     18301     IPv6 DSCP
so-0/1/1.0       69         8     IPv4 precedence
```

Rewrite Rule Mapping

```
user@R1> show class-of-service forwarding-table rewrite-rule mapping
Interface     Index   Table index  Type
so-0/1/1.0       68     58077     IPv6 DSCP
```

Scheduler Map

```
user@R1> show class-of-service scheduler-map diffserv-cos-map
Scheduler map: diffserv-cos-map, Index: 1094596010

  Scheduler: be-DATA-scheduler, Forwarding class: be-DATA-class, Index: 14343
    Transmit rate: 40 percent, Rate Limit: none, Buffer size: 40 percent,
    Priority: low
    Drop profiles:
      Loss priority   Protocol    Index    Name
      Low             non-TCP         1    <default-drop-profile>
      Low             TCP             1    <default-drop-profile>
      High            non-TCP         1    <default-drop-profile>
      High            TCP             1    <default-drop-profile>

  Scheduler: ef-FIN-scheduler, Forwarding class: ef-FIN-class, Index: 21707
    Transmit rate: 10 percent, Rate Limit: none, Buffer size: 10 percent,
    Priority: high
    Drop profiles:
      Loss priority   Protocol    Index    Name
      Low             non-TCP         1    <default-drop-profile>
      Low             TCP             1    <default-drop-profile>
      High            non-TCP         1    <default-drop-profile>
      High            TCP             1    <default-drop-profile>

  Scheduler: af-AV-scheduler, Forwarding class: af-AV-class, Index: 51704
    Transmit rate: 45 percent, Rate Limit: none, Buffer size: 45 percent,
    Priority: high
    Drop profiles:
      Loss priority   Protocol    Index    Name
      Low             non-TCP     61474    af-AV-normal
      Low             TCP         61474    af-AV-normal
      High            non-TCP     65199    af-AV-with-PLP
      High            TCP         65199    af-AV-with-PLP

  Scheduler: nc-CONTROL-scheduler, Forwarding class: nc-CONTROL-class, Index: 50404
    Transmit rate: 5 percent, Rate Limit: none, Buffer size: 5 percent,
    Priority: low
    Drop profiles:
      Loss priority   Protocol    Index    Name
      Low             non-TCP         1    <default-drop-profile>
      Low             TCP             1    <default-drop-profile>
      High            non-TCP         1    <default-drop-profile>
      High            TCP             1    <default-drop-profile>
```

Forwarding Table Scheduler Map

```
user@R1> show class-of-service forwarding-table scheduler-map
...
Interface: so-0/0/1 (Index: 141, Map index: -543019056, Map type: FINAL,
Num of queues: 4):
  Entry 0 (Scheduler index: 14343, Queue #: 0):
    Tx rate: 0 Kb (40%), Buffer size: 40 percent
Priority low
    PLP high: 1, PLP low: 1, TCP PLP high: 1, TCP PLP low: 1
  Entry 1 (Scheduler index: 21707, Queue #: 1):
    Tx rate: 0 Kb (10%), Buffer size: 10 percent
Priority high
    PLP high: 1, PLP low: 1, TCP PLP high: 1, TCP PLP low: 1
  Entry 2 (Scheduler index: 51704, Queue #: 2):
    Tx rate: 0 Kb (45%), Buffer size: 45 percent
Priority high
    PLP high: 65199, PLP low: 61474, TCP PLP high: 65199, TCP PLP low: 61474
  Entry 3 (Scheduler index: 50404, Queue #: 3):
    Tx rate: 0 Kb (5%), Buffer size: 5 percent
Priority low
    PLP high: 1, PLP low: 1, TCP PLP high: 1, TCP PLP low: 1

...
```

For More Information

For additional information about CoS using DiffServ and IPv6, see the following:

JUNOS Network Interfaces and Class of Service Configuration Guide

E. Elz, *A Compact Representation of IPv6 Addresses*, RFC 1924, April 1996.

K. Nichols *et al.*, *Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers*, RFC 2474, December 1998.

S. Blake *et al.*, *An Architecture for Differentiated Services*, RFC 2475, December 1998.

S. Deering *et al.*, *Internet Protocol, Version 6 (IPv6) Specification*, RFC 2640, December 1998.

D. Black, *Differentiated Service and Tunnels*, RFC 2983, October 2000.

D. Grossman, *New Terminology and Clarifications for DiffServ*, RFC 3260, April 2002.

K. Chan *et al.*, *Differentiated Services Quality of Service Policy Information Base*, RFC 3317, March 2003.

R. Hindon and S. Deering, *IP Version 6 Addressing Architecture*, RFC 3513, April 2003.

Revision History

2 February 2005—7.1R1 Release. Richard Hendricks.

6 October 2004—7.0R1 Release. Richard Hendricks.

6 July 2004—6.4R1 Release. Richard Hendricks.

5 April 2004—Initial document written, 6.3R1 Release. Walter Goralski.