Junos® Networking Technologies

DAY ONE: ADVANCED JUNOS CoS TROUBLESHOOTING COOKBOOK

Here's an advanced Junos Class of Service cookbook that emphasizes troubleshooting techniques and shows ways to find, fix, and commit complex CoS issues.

By Javed Syed & Nick Okerberg

DAY ONE: ADVANCED JUNOS CoS TROUBLESHOOTING COOKBOOK

In large Enterprise and Service Provider networks, dropped packets can quickly escalate into large scale issues, and to fix these Class of Service issues you have to know how and where to troubleshoot them. Here are eight Junos Class of Service issues and the how-to steps to fix them. Written by an Advance JTAC engineer and a Resident Engineer for one of world's largest Service Providers, these problem/solution/discussion recipes emphasize all the troubleshooting techniques available to the seasoned Junos engineer. Take this book into the lab and follow along.

Day One: Advanced Junos CoS Troubleshooting Cookbook is intended for network engineers, NOC engineers, design engineers, support staff, or planning staff that use Juniper Networks M, T, TX, or MX Series devices in a service provider scenario.

IT'S DAY ONE AND YOU HAVE A JOB TO DO, SO LEARN HOW TO:

- Better understand Class of Service troubleshooting techniques.
- Know and use special Junos commands needed for displaying Class of Service related outputs.
- Understand Junos Class of Service default behavior.

YOUR LAB SHOULD CONTAIN :

- One M, MX, or T Series router.
- An ethernet switch (with port mirroring capability).
- A server or workstation.

Published by Juniper Networks Books

ISBN 978-1936779659

51800

9 781936 779659

7100165

Junos® Networking Technologies

Day One: *Advanced Junos CoS Troubleshooting Cookbook*

By Javed Syed and Nick Okerberg

JUNIPer NETWORKS

Published by Juniper Networks Books
Author: Javed Syed, Nick Okerberg
Technical Reviewers: Alex Lee
Editor in Chief: Patrick Ames
Copyeditor and Proofer: Nancy Koerbel
J-Net Community Manager: Julie Wider

ISBN: 978-1-936779-65-9 (print)
Printed in the USA by Vervante Corporation.

ISBN: 978-1-936779-66-6 (ebook)

Version History: v2, April 15, 2013
3 4 5 6 7 8 9 10   #7100164-en

About the Authors
Javed Syed has over 17 years of experience in the networking industry, with the last 8 years focusing on service provider networks. He is CCIE # 2975, and JNCIP # 727, and also holds various other industry certifications. Javed is currently employed by Juniper Networks, where he functions as a solution engineer in the Professional Services organization. Javed previously worked at Cisco Systems and IBM.

Nick Okerberg is a resident engineer for Juniper Networks in the Professional Services group, supporting a large service provider in the United States. He has been employed by Juniper for about 7 years and has 10 years of experience in the networking industry. Nick previously did contract work for Juniper Networks and MSN. He holds the JNCIE-M #362 and JNCIE-ENT #304 certifications, among others.

Authors' Acknowledgments
Javed would like to thank his family for their support in allowing him the time to work on this book. Also, he would like to thank Nick Okerberg, Alex Lee, and Patrick Ames for their help in completing this project.

Nick would like to thank his family for their support and allowing him extra time to complete this project, Javed Syed for his help in writing this book, Alex Lee for all of his valuable feedback, and Patrick Ames for the assistance with editing.

This book is available in a variety of formats at: http://www.juniper.net/dayone. Send your suggestions, comments, and critiques by email to dayone@juniper.net.

Welcome to Day One

This book is part of a growing library of *Day One* books, produced and published by Juniper Networks Books.

Day One books were conceived to help you get just the information that you need on day one. The series covers Junos OS and Juniper Networks networking essentials with straightforward explanations, step-by-step instructions, and practical examples that are easy to follow.

The *Day One* library also includes a slightly larger and longer suite of *This Week* books, whose concepts and test bed examples are more similar to a weeklong seminar.

You can obtain either series, in multiple formats:

- Download a free PDF edition at http://www.juniper.net/dayone.
- Get the ebook edition for iPhones and iPads from the iTunes Store. Search for Juniper Networks Books.
- Get the ebook edition for any device that runs the Kindle app (Android, Kindle, iPad, PC, or Mac) by opening your device's Kindle app and going to the Kindle Store. Search for Juniper Networks Books.
- Purchase the paper edition at either Vervante Corporation (www.vervante.com) or Amazon (amazon.com) for between $12-$28, depending on page length.
- Note that Nook, iPad, and various Android apps can also view PDF files.
- If your device or ebook app uses .epub files, but isn't an Apple product, open iTunes and download the .epub file from the iTunes Store. You can now drag and drop the file out of iTunes onto your desktop and sync with your .epub device.

About Shell Commands and Junos

This book occasionally uses shell commands as part of its troubleshooting techniques. Note that using shell commands is not supported, or endorsed, but this book presumes you are a careful, prudent engineer who needs to solve a problem. Use only in a lab environment.

WARNING Do not run shell commands in production environments without JTAC supervision. Such commands may potentially cause impact to traffic.

Audience

The intended audience for this book is network engineers, NOC engineers, design engineers, support staff, or planning staff that use Juniper Networks M-series, T-series, TX, or MX-series devices in an enterprise or service provider environment.

What You Need to Know Before Reading

Before reading this book, you should be familiar with the basic administrative functions of the Junos operating system, including the ability to work with operational commands and to read, understand, and change Junos configurations. There are several books in the *Day One* library on learning Junos, at www.juniper.net/dayone.

This book makes a few assumptions about you, the reader:

- You have a basic understanding of the Internet Protocol (IP) versions 4 and 6.
- You have access to a lab with at least the following components: one M/MX/T-Series router, one ethernet switch (with port mirroring capability), and one server or workstation.

After Reading This Book, You'll Be Able To:

- Better understand Class of Service troubleshooting techniques.
- Know and use special Junos commands needed for displaying Class of Service related outputs.
- Understand Junos Class of Service default behavior.

Recipe 1

ICMP Ping Shows Latency

This recipe gives an example of a latency behavior on a PE-to-CE 10Gbps link on a Juniper MX480 routing platform. CLI commands are included to help show this behavior.

Problem

An ICMP ping shows intermittent latency on a directly connected link with the default Junos CoS configuration. Due to this behavior, some engineers may question the integrity of the link itself.

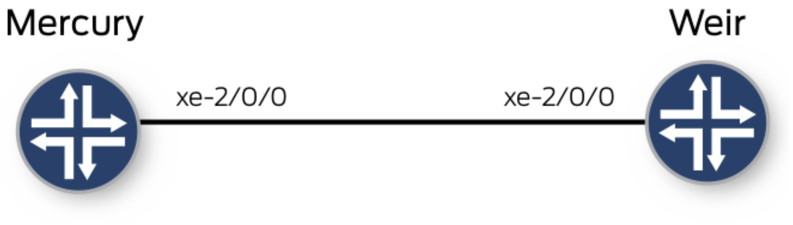
Such behavior could occur on any Junos routing platform with any link type. The higher the bandwidth of the link, the higher the odds of noticing intermittent latency on a directly connected link. This is because as the speed of the link increases, the lower the expected latency generally is for that link, and an intermittent latency problem would therefore be easier to notice.

Solution

It's common for engineers to use ping to test link availability in a network, however, sometimes there may be higher than expected latency seen when the ping is running. This is true even for a directly connected link that uses the default class-of-service configuration. It can happen in multiple scenarios, including a PE-to-CE link or a core PE-to-P link. Let's look at two quick examples.

In the lab, the router Mercury is a MX480 running 11.4R1.14 and the ping test is performed using ICHIP-based hardware. The ICHIP is the chipset used in the PFE components of certain routing platforms and hardware including:

- M120
- M320 E3 FPC Types
- MX240, MX480, MX960 DPCs



Example 1

And here is snippet of the ping on the directly connected interface:

```
lab@Mercury-re0> ping 10.1.1.2
<snip>
64 bytes from 10.1.1.2: icmp_seq=28 ttl=64 time=0.524 ms
64 bytes from 10.1.1.2: icmp_seq=29 ttl=64 time=0.503 ms
64 bytes from 10.1.1.2: icmp_seq=30 ttl=64 time=16.205 ms
64 bytes from 10.1.1.2: icmp_seq=31 ttl=64 time=0.543 ms
64 bytes from 10.1.1.2: icmp_seq=32 ttl=64 time=32.299 ms
64 bytes from 10.1.1.2: icmp_seq=33 ttl=64 time=0.533 ms
64 bytes from 10.1.1.2: icmp_seq=34 ttl=64 time=0.537 ms
<snip>
```

The snippet shows a ping to the remote end of the directly connected 10Gbps link. Notice that the round-trip time is under 1 ms for most of the Internet Control Message Protocol (ICMP) packets, but sometimes there is a spike far exceeding this time (shown in boldface). This intermittent increased delay could lead network engineers to question the reliability of the link reliability.

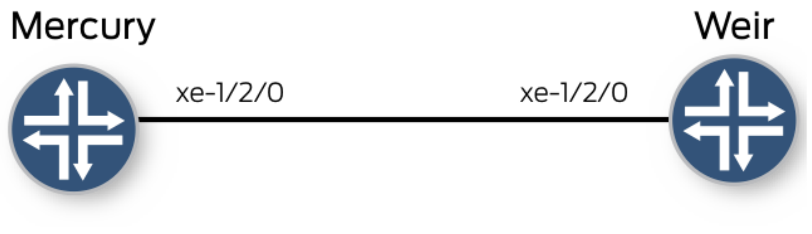
Let's show the same test on another chipset.

Example 2

In this example, the lab router Mercury is also an MX480 running 11.4R1.14, but the test is performed using trinity-based (Trio) hardware.

Trio is the chipset used on the following routing platforms and hardware:

- MX80
- MX240, MX480, MX960 MPCs



And here is a snippet:

```
lab@Mercury-re0> ping 10.1.1.2
<snip>
64 bytes from 10.1.1.2: icmp_seq=123 ttl=64 time=0.529 ms
64 bytes from 10.1.1.2: icmp_seq=124 ttl=64 time=0.526 ms
64 bytes from 10.1.1.2: icmp_seq=125 ttl=64 time=0.499 ms
64 bytes from 10.1.1.2: icmp_seq=126 ttl=64 time=0.532 ms
64 bytes from 10.1.1.2: icmp_seq=127 ttl=64 time=22.103 ms
64 bytes from 10.1.1.2: icmp_seq=128 ttl=64 time=60.453 ms
64 bytes from 10.1.1.2: icmp_seq=129 ttl=64 time=20.852 ms
64 bytes from 10.1.1.2: icmp_seq=130 ttl=64 time=58.912 ms
<snip>
```

The snippet shows a ping to the remote end of the 10Gbps link. Notice that the round-trip time is under 1 ms for some of the ICMP packets, but there are some responses that far exceed this time

Although ICMP is a great way to check for *link availability*, it is not a good way to test latency or delay on a Juniper Networks routing platform. One of the best and most accurate ways to test for latency is to simulate data plane traffic using a traffic generator.

Juniper Networks routing platform architecture separates the control plane from the data plane. There are various rate-limiting and prioritization functions within the PFE and Routing Engine. When a ping is done from the routing platform to the end device, it is using the control plane on the local router in order to generate the ICMP request packet and the control plane is used again when the ICMP reply packet is received from the end device. The ICMP messages are considered low priority within Junos, and the routing platform will respond to and process other higher priority messages, such as routing updates, before processing the ICMP messages.

The microkernel may introduce tens of milliseconds of processing delay to ICMP message handling. The delay is not uniform, meaning that *some* ICMP messages might be delayed while others may *not* be delayed. This matches the behavior observed here with the testing.

MORE? For more on ping see *This Week: A Packet Walkthrough on the M, MX, and T Series* at http://www.juniper.net/dayone.

Recipe 2

ICMP Drops

ICMP drops may be associated with a rapid ping of the directly connected device. This may lead engineers to believe that there is a problem with the link or circuit when in reality there may not be any issue at all.

Problem

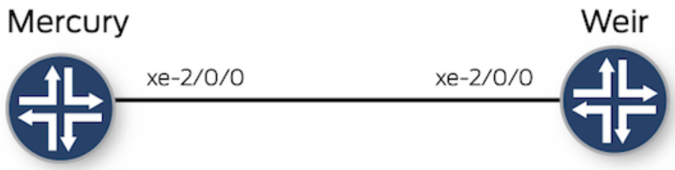
This behavior could happen on any Junos routing platform, but it is usually noticed more on higher speed interfaces such as 10Gbps links. That's because the higher speed interfaces can handle more packets at a faster rate, and Junos imposes a rate limit on low priority ICMP packets that are sent from the router or that are destined to the router. This is a default protection mechanism of the router to help prevent denial of service (DoS) attacks against the router.

This recipe shows us how to check various ICMP-related counters on the router to verify if such packets are being dropped due to a rate limit. Let's begin!

Solution

Sometimes it is possible to observe ICMP (Internet Control Message Protocol) packet drops when a rapid ping is used to the remote side of a directly connected interface while using the default class-of-service configuration. This could cause some engineers to believe that there is a link problem in the network, which may not be the case. Note that this could happen in multiple scenarios on a PE-to-CE link or on a core PE-to-P link.

NOTE In this example, the lab routers Mercury and Weir are MX480's running JUNOS 11.4R1.14. The test is performed using I-Chip-based hardware architecture. There is not any other traffic running.



A rapid ping is done from Mercury to the remote side of the link, Weir. There were a total of 58053 packets transmitted and only 58000 replies received by Mercury. So there was a loss of 53 packets during this test, as shown in the output below:

```
{master}
lab@Mercury-re0> ping 10.1.1.2 rapid count 100000
PING 10.1.1.2 (10.1.1.2): 56 data bytes
<snip>
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!.!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
^C
--- 10.1.1.2 ping statistics ---
58053 packets transmitted, 58000 packets received, 0% packet loss
round-trip min/avg/max/stddev = 0.265/0.553/53.249/1.440 ms
```

The output cites there is a 0% packet loss, but it is a rounded-down number. The actual loss was 53 packets, or about 0.99% of the transmitted amount.

Let's check on the remote side of this link, the Weir MX480 router:

```
lab@Weir-re0> show system statistics icmp
icmp:
         53 drops due to rate limit
         0 calls to icmp_error
         0 errors not generated because old message was icmp
         Output Histogram
                 58053 echo reply
         0 messages with bad code fields
         0 messages less than the minimum length
         0 messages with bad checksum
         0 messages with bad source address
         0 messages with bad length
```

```
0 echo drops with broadcast or multicast destination address
0 timestamp drops with broadcast or multicast destination address
Input Histogram
        1 destination unreachable
        58053 echo
58053 message responses generated
```

Here you can see there were ICMP drops on the remote side in the bolded output. This accounts for the 53 dropped packets. The `show system statistics icmp` command is used to display system-wide ICMP statistics on the router.

This is not a software bug, nor a problem with the link. Instead, the drops occurred on the MX480 Weir router because of the default rate-limiting of ICMP packets on Junos platforms. Weir is receiving all of the ICMP request packets and at the same time generating ICMP replies to send back to Mercury. However, during this process, it's hitting the default rate limit of ICMP within the kernel, which is on the routing-engine. The default ICMP rate limit on the system can be checked using the following shell command.

WARNING! Do not run shell commands in production without JTAC supervision. Such commands may potentially impact traffic.

```
lab@Weir-re0> start shell
% sysctl -a | grep "inet.icmp"
net.inet.icmp.maskrepl: 0
net.inet.icmp.bucketsize: 5
net.inet.icmp.tokenrate: 1000
net.inet.icmp.drop_redirect: 0
net.inet.icmp.log_redirect: 0
net.inet.icmp.bmcastecho: 1
% exit
exit
```

Notice in the output above that the ICMP rate is set to 1000, this is the default system value. If desired, the default value can be changed by using the `set system internet-options icmpv4-rate-limit configuration` command.

ICMP is not the best way to test the delay or capacity on the link. One of the best ways to test the delay or capacity is to attach a host or traffic generator to both sides. Then traffic can be sent through, or transit to, the routers. By taking a transit path through the router, there is not any default rate limiting that would limit the traffic.

Recipe 3

Queue Drops on an LSQ Interface

This recipe can be used to identify packet drops on an LSQ interface. It can also be used to help troubleshoot the source of the packet drops.

Problem

There are queue drops on a LSQ (Link Services IQ) interface even though all of the bandwidth in the link services bundle is not being used.

This problem could occur when the traffic flowing over the LSQ interface is bursty. The traffic rate received by the routing platform and sent out of the LSQ interface is rapidly fluctuating. For example, the traffic rate may be 2Mbps and rapidly decrease to 100Kbps before increasing back to 2Mbps again.

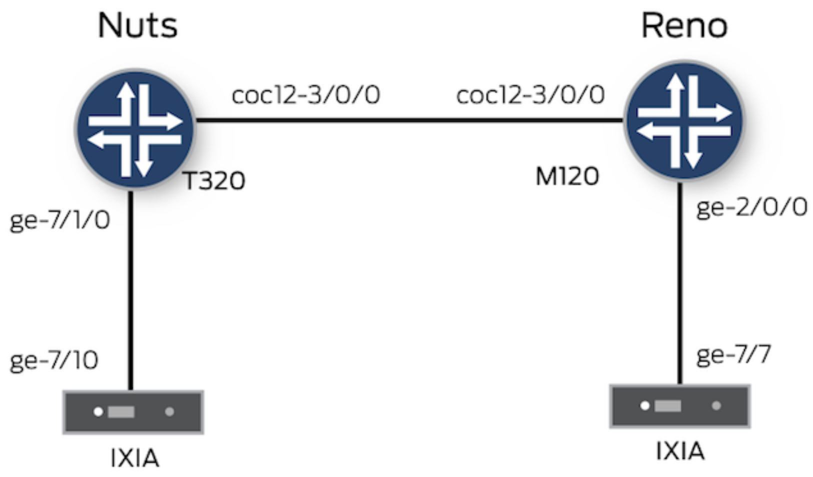
The bandwidth of a LSQ interface on a Juniper routing platform can vary because it's possible to bundle multiple interfaces together. For example, the bandwidth of a LSQ interface with two T1 interfaces would be about 3Mbps, while the bandwidth with three T1 interfaces would be about 4.6Mbps.

Solution

You might notice dropped packets on an LSQ interface even if the bundle is not being fully utilized, and this may seem odd at first, because usually dropped packets are due to congestion and happen when the link is fully utilized. But here the drops appear to be happening even when the link is not being fully utilized.

Let's replicate the queue drops in our lab and analyze the results.

The lab routers Nuts and Reno are a T320 and a M120, respectively. They are both running Junos 11.4R1.14 and they are connected to an IXIA Traffic Generator.



In the lab topology, the IXIA traffic generator is sending traffic from port ge-7/10 to ge-7/7. There are two different traffic streams being used:

Traffic stream #1:

- Source IP address is 192.168.1.2.
- Destination IP address range is 192.168.2.100 – 192.168.2.200.
- Protocol UDP, source and destination port 63.
- DSCP bits 000000.
- Frame size is random, between 64 and 1500 bytes.
- Sends 125 packets in 5 microseconds then proceeds to Traffic stream #2.

Traffic stream #2:

- Source IP address is 192.168.1.2.
- Destination IP address range is 192.168.2.10 – 192.168.2.200.
- Protocol UDP, source and destination port 63.

- DSCP bits 000000.
- Frame size is random, between 64 and 1500 bytes.
- Sends 630 packets at about a 400Kbps rate, which lasts about 10 seconds. Then proceeds to Traffic stream #1.

The device under test (DUT) in this scenario is the T320 Nuts router. Specifically, the egress side of the LSQ interface on the T320 that connects to the M120.

Here is the configuration of the LSQ interface and the associated T1 interfaces. They were created from a Channelized OC12 port. Notice that there are three member T1 interfaces that are part of the LSQ bundle.

```
interfaces {
   lsq-1/1/0 {
      per-unit-scheduler;
      unit 0 {
         encapsulation multilink-ppp;
         family inet {
            address 10.1.1.1/30;
         }
      }
   }
   coc12-3/0/0 {
      partition 1 oc-slice 1 interface-type coc1;
   }
   coc1-3/0/0:1 {
      partition 1-4 interface-type t1;
   }
   t1-3/0/0:1:1 {
      unit 0 {
         family mlppp {
            bundle lsq-1/1/0.0;
         }
      }
   }
   t1-3/0/0:1:2 {
      unit 0 {
         family mlppp {
            bundle lsq-1/1/0.0;
         }
      }
   }
   t1-3/0/0:1:3 {
      unit 0 {
         family mlppp {
            bundle lsq-1/1/0.0;
         }
      }
   }
   <truncated>
}
```

Next, the `class-of-service` configuration on the T320 Nuts router is shown below. The LSQ bundle has the user-defined scheduler-map “MAP_1” applied to it.

```
class-of-service {
   classifiers {
      dscp CLASSIFIER_1 {
         forwarding-class DATA {
            loss-priority low code-points 000000;
         }
         forwarding-class APPS {
            loss-priority low code-points 001010;
         }
         forwarding-class VOICE {
            loss-priority high code-points 101110;
         }
         forwarding-class NC {
            loss-priority high code-points 110000;
         }
      }
   }
   forwarding-classes {
      queue 0 DATA;
      queue 1 VOICE;
      queue 2 APPS;
      queue 3 NC;
   }
   interfaces {
      lsq-1/1/0 {
         unit 0 {
            scheduler-map MAP_1;
         }
      }
      ge-7/1/0 {
         unit 0 {
            classifiers {
               dscp CLASSIFIER_1;
            }
         }
      }
   }
   scheduler-maps {
      MAP_1 {
         forwarding-class DATA scheduler DATA-BE;
         forwarding-class VOICE scheduler VOICE-EF;
         forwarding-class APPS scheduler APPS-AF;
         forwarding-class NC scheduler NC;
      }
   }
   schedulers {
      DATA-BE {
         transmit-rate percent 70;
         buffer-size percent 70;
         priority low;
```

```
        }
        VOICE-EF {
            transmit-rate percent 20;
            buffer-size percent 20;
            priority high;
        }
        APPS-AF {
            transmit-rate percent 5;
            buffer-size percent 5;
            priority medium-low;
        }
        NC {
            transmit-rate percent 5;
            buffer-size percent 5;
            priority strict-high;
        }
    }
}

[edit]
lab@Nuts-re0# run show class-of-service interface lsq-1/1/0
May 01 15:36:15
Physical interface: lsq-1/1/0, Index: 204
Queues supported: 8, Queues in use: 4
  Scheduler map: <default-chassis>, Index: 4
  Chassis scheduler map: <default-chassis>, Index: 4
  Congestion-notification: Disabled

  Logical interface: lsq-1/1/0.0, Index: 92
    Object                  Name                   Type                    Index
    Scheduler-map           MAP_1                  Output                   53024
    Classifier              ipprec-compatibility   ip                          13
```

If you look at the lsq-1/1/0 interface statistics below, they show that the traffic is correctly utilizing all three of the member links. The frames per second (fps) value for the three member links are close to being equal as well.

```
[edit]
lab@Nuts-re0# run show interfaces lsq-1/1/0 extensive | no-more
May 01 15:45:24
Physical interface: lsq-1/1/0, Enabled, Physical link is Up
  Interface index: 204, SNMP ifIndex: 603, Generation: 325
  Link-level type: LinkService, MTU: 1504
  Device flags   : Present Running
  Interface flags: Point-To-Point SNMP-Traps Internal: 0x4000
  Last flapped   : 2012-05-01 14:23:25 PDT (01:21:59 ago)
  Statistics last cleared: 2012-05-01 15:35:52 PDT (00:09:32 ago)
  Traffic statistics:
   Input  bytes  :                5200                    0 bps
   Output bytes  :            33457321               381896 bps
   Input  packets:                  65                    0 pps
   Output packets:               43447                   63 pps
   IPv6 transit statistics:
        Input  bytes  :               0
        Output bytes  :               0
        Input  packets:               0
        Output packets:               0
      Frame exceptions:
        Oversized frames              0
        Errored input frames          0
        Input on disabled link/bundle     0
        Output for disabled link/bundle   0
        Queuing drops                 0
      Buffering exceptions:
        Packet data buffer overflow      0
        Fragment data buffer overflow    0
      Assembly exceptions:
        Fragment timeout              0
        Missing sequence number         0
        Out-of-order sequence number     0
        Out-of-range sequence number     0
      Hardware errors (sticky):
        Data memory error             0
        Control memory error           0
      Egress queues: 8 supported, 4 in use
      Queue counters:      Queued packets  Transmitted packets     Dropped packets
        0 DATA                      0                  0                  0
        1 VOICE                     0                  0                  0
        2 APPS                      0                  0                  0
        3 NC                        0                  0                  0
      Queue number:       Mapped forwarding classes
        0                DATA
        1                VOICE
        2                APPS
        3                NC

      Logical interface lsq-1/1/0.0 (Index 92) (SNMP ifIndex 608) (Generation 299)
        Flags: Point-To-Point SNMP-Traps 0x4000 Encapsulation: Multilink-PPP
        Last flapped: 2012-05-01 15:26:08 PDT (00:19:16 ago)
        Bandwidth: 4608kbps
        Bundle links information:
          Active bundle links      3
          Removed bundle links     0
          Disabled bundle links    0
        Bundle options:
          MRRU                           1504
          Remote MRRU                     1504
          Drop timer period               500
          Inner PPP Protocol field compression enabled
          Sequence number format          long (24 bits)
          Fragmentation threshold          0
          Links needed to sustain bundle      1
          Multilink classes               0
          Link layer overhead             4.0 %
        Bundle status:
          Received sequence number       0x8d
          Transmit sequence number       0x14fcc
          Packet drops                  0 (0 bytes)
          Fragment drops                0 (0 bytes)
          MRRU exceeded                 0
          Fragment timeout              0
          Missing sequence number       0
          Out-of-order sequence number   0
          Out-of-range sequence number   0
          Packet data buffer overflow    0
          Fragment data buffer overflow  0
        Statistics       Frames      fps       Bytes         bps
        Bundle:
          Multilink:
            Input :          65        0        5590           0
            Output:       43312       61    33830752      382064
          Network:
            Input :          65        0        5200           0
            Output:       43447       63    33457321      381552
        Link:
          t1-3/0/0:1:1.0
            Up time: 00:19:16
            Input :          22        0        1892           0
            Output:       14508       20    11271159      124368
          t1-3/0/0:1:2.0
            Up time: 00:19:16
            Input :          21        0        1806           0
            Output:       14438       22    11291009      130920
          t1-3/0/0:1:3.0
            Up time: 00:19:16
            Input :          22        0        1892           0
            Output:       14366       19    11268584      126776
        Multilink detail statistics:
        Bundle:
          Fragments:
            Input :           0        0           0           0
            Output:           0        0           0           0
          Non-fragments:
            Input :          65        0        5590           0
            Output:       43067       64    33550359      434496
          LFI:
            Input :           0        0           0           0
            Output:           0        0           0           0
       NCP state: inet: Opened, inet6: Not-configured, iso: Not-configured, mpls: Not-
     configured
        Protocol inet, MTU: 1500, Generation: 450, Route table: 0
          Flags: Sendbcast-pkt-to-re
          Addresses, Flags: Is-Preferred Is-Primary
            Destination: 10.1.1.0/30, Local: 10.1.1.1, Broadcast: Unspecified, Generation:
     285
```

Next, the following sequence of CLI outputs shows the problem. (This CLI command would need to be run several times and inspected in order to see if there are any drops incrementing.) The `show interfaces queue lsq-1/1/0.0` command reveals that there are RED dropped packets incrementally growing in queue 0. At the same time, notice the transmitted packets in queue 0 are always lower than the total amount of available bandwidth for the three T1 interfaces, giving the impression that the traffic rate being sent out of the LSQ interface is lower than the capacity of the LSQ bundle;the LSQ bundle should be able to handle the amount of traffic that is being sent.

As an alternative, the `show interfaces extensive lsq-1/1/0` command will also show packet drops, but the `show interfaces queue lsq-1/1/0.0` command is used here because it gives the packet rates for the amount of traffic queued and transmitted.

```
[edit]
lab@Nuts-re0# run show interfaces queue lsq-1/1/0.0
May 01 15:48:58
  Logical interface lsq-1/1/0.0 (Index 92) (SNMP ifIndex 608)
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Burst size: 0
Queue: 0, Forwarding classes: DATA
  Queued:
    Packets              :                59552                   63 pps
    Bytes                :             45972016               366824 bps
  Transmitted:
    Packets              :                59159                   63 pps
    Bytes                :             45659748               366824 bps
    Tail-dropped packets :                    0                    0 pps
    RED-dropped packets  :                  393                    0 pps
     Low, non-TCP        :                  393                    0 pps
     Low, TCP            :                    0                    0 pps
     High, non-TCP       :                    0                    0 pps
     High, TCP           :                    0                    0 pps
    RED-dropped bytes    :               312268                    0 bps
     Low, non-TCP        :               312268                    0 bps
     Low, TCP            :                    0                    0 bps
     High, non-TCP       :                    0                    0 bps
     High, TCP           :                    0                    0 bps
Queue: 1, Forwarding classes: VOICE
  Queued:
    Packets              :                    0                    0 pps
    Bytes                :                    0                    0 bps
  Transmitted:
    Packets              :                    0                    0 pps
    Bytes                :                    0                    0 bps
    Tail-dropped packets :                    0                    0 pps
    RED-dropped packets  :                    0                    0 pps
     Low, non-TCP        :                    0                    0 pps
     Low, TCP            :                    0                    0 pps
     High, non-TCP       :                    0                    0 pps
     High, TCP           :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
     Low, non-TCP        :                    0                    0 bps
     Low, TCP            :                    0                    0 bps
     High, non-TCP      :                  0                     0 bps
     High, TCP          :                  0                     0 bps
Queue: 2, Forwarding classes: APPS
  Queued:
    Packets             :                  0                     0 pps
    Bytes               :                  0                     0 bps
  Transmitted:
    Packets             :                  0                     0 pps
    Bytes               :                  0                     0 bps
    Tail-dropped packets :                  0                     0 pps
    RED-dropped packets  :                  0                     0 pps
     Low, non-TCP       :                  0                     0 pps
     Low, TCP           :                  0                     0 pps
     High, non-TCP      :                  0                     0 pps
     High, TCP          :                  0                     0 pps
    RED-dropped bytes    :                  0                     0 bps
     Low, non-TCP       :                  0                     0 bps
     Low, TCP           :                  0                     0 bps
     High, non-TCP      :                  0                     0 bps
     High, TCP          :                  0                     0 bps
Queue: 3, Forwarding classes: NC
  Queued:
    Packets             :                 90                     0 pps
    Bytes               :               7200                     0 bps
  Transmitted:
    Packets             :                 90                     0 pps
    Bytes               :               7200                     0 bps
    Tail-dropped packets :                  0                     0 pps
    RED-dropped packets  :                  0                     0 pps
     Low, non-TCP       :                  0                     0 pps
     Low, TCP           :                  0                     0 pps
     High, non-TCP      :                  0                     0 pps
     High, TCP          :                  0                     0 pps
    RED-dropped bytes    :                  0                     0 bps
     Low, non-TCP       :                  0                     0 bps
     Low, TCP           :                  0                     0 bps
     High, non-TCP      :                  0                     0 bps
     High, TCP          :                  0                     0 bps

[edit]
lab@Nuts-re0# run show interfaces queue lsq-1/1/0.0
May 01 15:49:04
  Logical interface lsq-1/1/0.0 (Index 92) (SNMP ifIndex 608)
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Burst size: 0
Queue: 0, Forwarding classes: DATA
  Queued:
    Packets             :              60056                    63 pps
    Bytes               :           46371598                428872 bps
  Transmitted:
    Packets             :              59662                    63 pps
    Bytes               :           46058659                428872 bps
    Tail-dropped packets :                  0                     0 pps
    RED-dropped packets  :             **394**                0 pps
     Low, non-TCP       :             394                0 pps
     Low, TCP           :             0                0 pps
     High, non-TCP      :              0                0 pps
     High, TCP          :             0                0 pps
    RED-dropped bytes    :            312939                0 bps
     Low, non-TCP       :           312939               0 bps
     Low, TCP           :             0                0 bps
     High, non-TCP      :              0                0 bps
     High, TCP          :             0                0 bps
Queue: 1, Forwarding classes: VOICE
  Queued:
    Packets             :             0                0 pps
    Bytes               :             0                0 bps
  Transmitted:
    Packets             :             0                0 pps
    Bytes               :             0                0 bps
    Tail-dropped packets :              0                0 pps
    RED-dropped packets  :              0                0 pps
     Low, non-TCP       :              0                0 pps
     Low, TCP           :             0                0 pps
     High, non-TCP      :              0                0 pps
     High, TCP          :             0                0 pps
    RED-dropped bytes    :              0                0 bps
     Low, non-TCP       :              0                0 bps
     Low, TCP           :             0                0 bps
     High, non-TCP      :              0                0 bps
     High, TCP          :             0                0 bps
Queue: 2, Forwarding classes: APPS
  Queued:
    Packets             :             0                0 pps
    Bytes               :             0                0 bps
  Transmitted:
    Packets             :             0                0 pps
    Bytes               :             0                0 bps
    Tail-dropped packets :              0                0 pps
    RED-dropped packets  :              0                0 pps
     Low, non-TCP       :              0                0 pps
     Low, TCP           :             0                0 pps
     High, non-TCP      :              0                0 pps
     High, TCP          :             0                0 pps
    RED-dropped bytes    :              0                0 bps
     Low, non-TCP       :              0                0 bps
     Low, TCP           :             0                0 bps
     High, non-TCP      :              0                0 bps
     High, TCP          :             0                0 bps
Queue: 3, Forwarding classes: NC
  Queued:
    Packets             :            91                1 pps
    Bytes               :           7280             640 bps
  Transmitted:
    Packets             :            91                1 pps
    Bytes               :           7280             640 bps
    Tail-dropped packets :              0                0 pps
    RED-dropped packets  :                   0                   0 pps
     Low, non-TCP       :                  0                  0 pps
     Low, TCP           :                  0                  0 pps
     High, non-TCP      :                  0                  0 pps
     High, TCP          :                  0                  0 pps
    RED-dropped bytes    :                   0                   0 bps
     Low, non-TCP       :                  0                  0 bps
     Low, TCP           :                  0                  0 bps
     High, non-TCP      :                  0                  0 bps
     High, TCP          :                  0                  0 bps

[edit]
lab@Nuts-re0# run show interfaces queue lsq-1/1/0.0
May 01 15:49:11
  Logical interface lsq-1/1/0.0 (Index 92) (SNMP ifIndex 608)
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Burst size: 0
Queue: 0, Forwarding classes: DATA
  Queued:
    Packets             :               60624                   64 pps
    Bytes               :            46826101               358056 bps
  Transmitted:
    Packets             :               60222                  100 pps
    Bytes               :            46503808               620968 bps
    Tail-dropped packets :                  0                    0 pps
    RED-dropped packets  :                402                    0 pps
     Low, non-TCP       :                402                  0 pps
     Low, TCP           :                  0                  0 pps
     High, non-TCP      :                  0                  0 pps
     High, TCP          :                  0                  0 pps
    RED-dropped bytes    :             322293                    0 bps
     Low, non-TCP       :             322293                   0 bps
     Low, TCP           :                  0                  0 bps
     High, non-TCP      :                  0                  0 bps
     High, TCP          :                  0                  0 bps
Queue: 1, Forwarding classes: VOICE
  Queued:
    Packets             :                  0                  0 pps
    Bytes               :                  0                  0 bps
  Transmitted:
    Packets             :                  0                  0 pps
    Bytes               :                  0                  0 bps
    Tail-dropped packets :                  0                    0 pps
    RED-dropped packets  :                   0                   0 pps
     Low, non-TCP       :                  0                  0 pps
     Low, TCP           :                  0                  0 pps
     High, non-TCP      :                  0                  0 pps
     High, TCP          :                  0                  0 pps
    RED-dropped bytes    :                   0                   0 bps
     Low, non-TCP       :                  0                  0 bps
     Low, TCP           :                  0                  0 bps
     High, non-TCP      :                  0                  0 bps
     High, TCP          :                  0                  0 bps
```

```
Queue: 2, Forwarding classes: APPS
  Queued:
    Packets              :                 0                 0 pps
    Bytes                :                 0                 0 bps
  Transmitted:
    Packets              :                 0                 0 pps
    Bytes                :                 0                 0 bps
    Tail-dropped packets :                 0                 0 pps
    RED-dropped packets  :                 0                 0 pps
     Low, non-TCP        :                 0                 0 pps
     Low, TCP            :                 0                 0 pps
     High, non-TCP       :                 0                 0 pps
     High, TCP           :                 0                 0 pps
    RED-dropped bytes    :                 0                 0 bps
     Low, non-TCP        :                 0                 0 bps
     Low, TCP            :                 0                 0 bps
     High, non-TCP       :                 0                 0 bps
     High, TCP           :                 0                 0 bps
Queue: 3, Forwarding classes: NC
  Queued:
    Packets              :                91                 0 pps
    Bytes                :              7280                 0 bps
  Transmitted:
    Packets              :                91                 0 pps
    Bytes                :              7280                 0 bps
    Tail-dropped packets :                 0                 0 pps
    RED-dropped packets  :                 0                 0 pps
     Low, non-TCP        :                 0                 0 pps
     Low, TCP            :                 0                 0 pps
     High, non-TCP       :                 0                 0 pps
     High, TCP           :                 0                 0 pps
    RED-dropped bytes    :                 0                 0 bps
     Low, non-TCP        :                 0                 0 bps
     Low, TCP            :                 0                 0 bps
     High, non-TCP       :                 0                 0 bps
     High, TCP           :                 0                 0 bps
```

So, the lab output shows that the drops are occurring on the LSQ bundle when the bandwidth appears to be well below the total amount available. The `q-pic-large-buffer` command can be added to give the PIC a larger delay buffer, and in some scenarios, this could help reduce the amount of drops.

WARNING Committing this `q-pic-large-buffer` knob causes the PIC to reset.

```
[edit]
lab@Nuts-re0# set chassis fpc 3 pic 0 q-pic-large-buffer

[edit]
lab@Nuts-re0# show chassis
May 01 15:52:40
fpc 1 {
   pic 1 {
        adaptive-services {
            service-package layer-2;
        }
    }
}
fpc 3 {
    pic 0 {
        q-pic-large-buffer;
    }
}

[edit]
lab@Nuts-re0# commit
May 01 15:52:46
commit complete

[edit]
lab@Nuts-re0# run show chassis pic fpc-slot 3 pic-slot 0
May 01 15:52:54
FPC slot 3, PIC slot 0 information:
  Type                             1x CHOC12 IQ SONET, SMIR
  State                            Online
  PIC version                   1.22
  Uptime                          5 seconds
```

Next, clear the interface statistics so you can start from a fresh state to see if the drops are still happening:

```
[edit]
lab@Nuts-re0# run clear interfaces statistics all
```

Now, after adding the q-pic-large-buffer knob, the problem still seems to be happening as shown here:

```
[edit]
lab@Nuts-re0# run show interfaces queue lsq-1/1/0.0 forwarding-class DATA
May 01 15:54:27
  Logical interface lsq-1/1/0.0 (Index 92) (SNMP ifIndex 608)
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Burst size: 0
Queue: 0, Forwarding classes: DATA
  Queued:
    Packets              :              2398              64 pps
    Bytes                :           1859727          387768 bps
  Transmitted:
    Packets              :              2373              64 pps
    Bytes                :           1838110          387768 bps
    Tail-dropped packets :                 0               0 pps
    RED-dropped packets  :                25               0 pps
     Low, non-TCP        :                25               0 pps
     Low, TCP            :                 0               0 pps
     High, non-TCP       :                 0               0 pps
     High, TCP          :                  0                  0 pps
    RED-dropped bytes    :               21617                  0 bps
     Low, non-TCP        :              21617                 0 bps
     Low, TCP           :                 0                  0 bps
     High, non-TCP       :                 0                  0 bps
     High, TCP          :                  0                  0 bps

[edit]
lab@Nuts-re0# run show interfaces queue lsq-1/1/0.0 forwarding-class DATA
May 01 15:54:28
  Logical interface lsq-1/1/0.0 (Index 92) (SNMP ifIndex 608)
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Burst size: 0
Queue: 0, Forwarding classes: DATA
  Queued:
    Packets             :               2461                 63 pps
    Bytes               :            1912014             418296 bps
  Transmitted:
    Packets             :               2436                 63 pps
    Bytes               :            1890397             418296 bps
    Tail-dropped packets :                  0                  0 pps
    RED-dropped packets  :                 25                  0 pps
     Low, non-TCP        :                25                 0 pps
     Low, TCP           :                 0                  0 pps
     High, non-TCP       :                 0                  0 pps
     High, TCP          :                  0                  0 pps
    RED-dropped bytes    :               21617                  0 bps
     Low, non-TCP        :              21617                 0 bps
     Low, TCP           :                 0                  0 bps
     High, non-TCP       :                 0                  0 bps
     High, TCP          :                  0                  0 bps

[edit]
lab@Nuts-re0# run show interfaces queue lsq-1/1/0.0 forwarding-class DATA
May 01 15:54:30
  Logical interface lsq-1/1/0.0 (Index 92) (SNMP ifIndex 608)
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Burst size: 0
Queue: 0, Forwarding classes: DATA
  Queued:
    Packets             :               2712                 63 pps
    Bytes               :            2108823             390184 bps
  Transmitted:
    Packets             :               2680                 86 pps
    Bytes               :            2080619             534720 bps
    Tail-dropped packets :                  0                  0 pps
    RED-dropped packets  :                 32                  0 pps
     Low, non-TCP        :                32                 0 pps
     Low, TCP           :                 0                  0 pps
     High, non-TCP       :                 0                  0 pps
     High, TCP          :                  0                  0 pps
    RED-dropped bytes    :               28204                  0 bps
```

```
     Low, non-TCP      :             28204                0 bps
     Low, TCP          :                 0              0 bps
     High, non-TCP     :                 0              0 bps
     High, TCP         :                 0              0 bps

[edit]
lab@Nuts-re0# run show interfaces queue lsq-1/1/0.0 forwarding-class DATA
May 01 15:54:30
  Logical interface lsq-1/1/0.0 (Index 92) (SNMP ifIndex 608)
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Burst size: 0
Queue: 0, Forwarding classes: DATA
  Queued:
    Packets           :              2712             63 pps
    Bytes             :           2108823         390184 bps
  Transmitted:
    Packets           :              2680             86 pps
    Bytes             :           2080619         534720 bps
    Tail-dropped packets :                 0              0 pps
    RED-dropped packets  :                32              0 pps
     Low, non-TCP      :                32              0 pps
     Low, TCP          :                 0              0 pps
     High, non-TCP     :                 0              0 pps
     High, TCP         :                 0              0 pps
    RED-dropped bytes   :             28204                0 bps
     Low, non-TCP      :             28204                0 bps
     Low, TCP          :                 0              0 bps
     High, non-TCP     :                 0              0 bps
     High, TCP         :                 0              0 bps

[edit]
lab@Nuts-re0# run show interfaces queue lsq-1/1/0.0 forwarding-class DATA
May 01 15:54:31
  Logical interface lsq-1/1/0.0 (Index 92) (SNMP ifIndex 608)
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Burst size: 0
Queue: 0, Forwarding classes: DATA
  Queued:
    Packets           :              2775             63 pps
    Bytes             :           2152835         352096 bps
  Transmitted:
    Packets           :              2743             63 pps
    Bytes             :           2124631         352096 bps
    Tail-dropped packets :                 0              0 pps
    RED-dropped packets  :                32              0 pps
     Low, non-TCP      :                32              0 pps
     Low, TCP          :                 0              0 pps
     High, non-TCP     :                 0              0 pps
     High, TCP         :                 0              0 pps
    RED-dropped bytes   :             28204                0 bps
     Low, non-TCP      :             28204                0 bps
     Low, TCP          :                 0              0 bps
     High, non-TCP      :                   0                  0 bps
     High, TCP         :                   0                  0 bps

[edit]
lab@Nuts-re0# run show interfaces queue lsq-1/1/0.0 forwarding-class DATA
 May 01 15:54:32
   Logical interface lsq-1/1/0.0 (Index 92) (SNMP ifIndex 608)
 Forwarding classes: 16 supported, 4 in use
 Egress queues: 8 supported, 4 in use
 Burst size: 0
 Queue: 0, Forwarding classes: DATA
   Queued:
    Packets           :              2902                63 pps
    Bytes             :           2252970            436472 bps
   Transmitted:
    Packets           :              2870                63 pps
    Bytes             :           2224766            436472 bps
    Tail-dropped packets :                 0                 0 pps
    RED-dropped packets  :                32                 0 pps
     Low, non-TCP       :                32                0 pps
     Low, TCP          :                 0                0 pps
     High, non-TCP      :                 0                0 pps
     High, TCP         :                 0                0 pps
    RED-dropped bytes    :             28204                 0 bps
     Low, non-TCP       :             28204                0 bps
     Low, TCP          :                 0                0 bps
     High, non-TCP      :                 0                0 bps
     High, TCP         :                 0                0 bps

 [edit]
 lab@Nuts-re0# run show interfaces queue lsq-1/1/0.0 forwarding-class DATA
 May 01 15:54:33
   Logical interface lsq-1/1/0.0 (Index 92) (SNMP ifIndex 608)
 Forwarding classes: 16 supported, 4 in use
 Egress queues: 8 supported, 4 in use
 Burst size: 0
 Queue: 0, Forwarding classes: DATA
   Queued:
    Packets           :              2902                63 pps
    Bytes             :           2252970            436472 bps
   Transmitted:
    Packets           :              2870                63 pps
    Bytes             :           2224766            436472 bps
    Tail-dropped packets :                 0                 0 pps
    RED-dropped packets  :                32                 0 pps
     Low, non-TCP       :                32                0 pps
     Low, TCP          :                 0                0 pps
     High, non-TCP      :                 0                0 pps
     High, TCP         :                 0                0 pps
    RED-dropped bytes    :             28204                 0 bps
     Low, non-TCP       :             28204                0 bps
     Low, TCP          :                 0                0 bps
     High, non-TCP      :                 0                0 bps
     High, TCP         :                 0                0 bps
```

```
[edit]
lab@Nuts-re0# run show interfaces queue lsq-1/1/0.0 forwarding-class DATA
May 01 15:54:34
  Logical interface lsq-1/1/0.0 (Index 92) (SNMP ifIndex 608)
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Burst size: 0
Queue: 0, Forwarding classes: DATA
  Queued:
    Packets              :                  2965                    63 pps
    Bytes                :               2304010                408320 bps
  Transmitted:
    Packets              :                  2933                    63 pps
    Bytes                :               2275806                408320 bps
    Tail-dropped packets :                     0                     0 pps
    RED-dropped packets  :                    32                     0 pps
     Low, non-TCP        :                    32                     0 pps
     Low, TCP            :                     0                     0 pps
     High, non-TCP       :                     0                     0 pps
     High, TCP           :                     0                     0 pps
    RED-dropped bytes    :                 28204                     0 bps
     Low, non-TCP        :                 28204                     0 bps
     Low, TCP            :                     0                     0 bps
     High, non-TCP       :                     0                     0 bps
     High, TCP           :                     0                     0 bps

[edit]
lab@Nuts-re0# run show interfaces queue lsq-1/1/0.0 forwarding-class DATA
May 01 15:54:35
  Logical interface lsq-1/1/0.0 (Index 92) (SNMP ifIndex 608)
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Burst size: 0
Queue: 0, Forwarding classes: DATA
  Queued:
    Packets              :                  3028                    63 pps
    Bytes                :               2353368                394864 bps
  Transmitted:
    Packets              :                  2996                    63 pps
    Bytes                :               2325164                394864 bps
    Tail-dropped packets :                     0                     0 pps
    RED-dropped packets  :                    32                     0 pps
     Low, non-TCP        :                    32                     0 pps
     Low, TCP            :                     0                     0 pps
     High, non-TCP       :                     0                     0 pps
     High, TCP           :                     0                     0 pps
    RED-dropped bytes    :                 28204                     0 bps
     Low, non-TCP        :                 28204                     0 bps
     Low, TCP            :                     0                     0 bps
     High, non-TCP       :                     0                     0 bps
     High, TCP           :                     0                     0 bps

[edit]
```

```
lab@Nuts-re0# run show interfaces queue lsq-1/1/0.0 forwarding-class DATA
May 01 15:54:35
  Logical interface lsq-1/1/0.0 (Index 92) (SNMP ifIndex 608)
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Burst size: 0
Queue: 0, Forwarding classes: DATA
  Queued:
    Packets              :                3028                    63 pps
    Bytes                :             2353368                394864 bps
  Transmitted:
    Packets              :                2996                    63 pps
    Bytes                :             2325164                394864 bps
    Tail-dropped packets :                   0                     0 pps
    RED-dropped packets  :                  32                     0 pps
     Low, non-TCP        :                  32                     0 pps
     Low, TCP            :                   0                     0 pps
     High, non-TCP       :                   0                     0 pps
     High, TCP           :                   0                     0 pps
    RED-dropped bytes    :               28204                     0 bps
     Low, non-TCP        :               28204                     0 bps
     Low, TCP            :                   0                     0 bps
     High, non-TCP       :                   0                     0 bps
     High, TCP           :                   0                     0 bps

[edit]
lab@Nuts-re0# run show interfaces queue lsq-1/1/0.0 forwarding-class DATA
May 01 15:54:36
  Logical interface lsq-1/1/0.0 (Index 92) (SNMP ifIndex 608)
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Burst size: 0
Queue: 0, Forwarding classes: DATA
  Queued:
    Packets              :                3091                    63 pps
    Bytes                :             2401530                385296 bps
  Transmitted:
    Packets              :                3059                    63 pps
    Bytes                :             2373326                385296 bps
    Tail-dropped packets :                   0                     0 pps
    RED-dropped packets  :                  32                     0 pps
     Low, non-TCP        :                  32                     0 pps
     Low, TCP            :                   0                     0 pps
     High, non-TCP       :                   0                     0 pps
     High, TCP           :                   0                     0 pps
    RED-dropped bytes    :               28204                     0 bps
     Low, non-TCP        :               28204                     0 bps
     Low, TCP            :                   0                     0 bps
     High, non-TCP       :                   0                     0 bps
     High, TCP           :                   0                     0 bps

[edit]
lab@Nuts-re0# run show interfaces queue lsq-1/1/0.0 forwarding-class DATA
May 01 15:54:37
```

```
  Logical interface lsq-1/1/0.0 (Index 92) (SNMP ifIndex 608)
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Burst size: 0
Queue: 0, Forwarding classes: DATA
  Queued:
    Packets              :                3155                64 pps
    Bytes                :             2448598            376544 bps
  Transmitted:
    Packets              :                3123                64 pps
    Bytes                :             2420394            376544 bps
    Tail-dropped packets :                   0                 0 pps
    RED-dropped packets  :                  32                 0 pps
     Low, non-TCP        :                  32                 0 pps
     Low, TCP            :                   0                 0 pps
     High, non-TCP       :                   0                 0 pps
     High, TCP           :                   0                 0 pps
    RED-dropped bytes    :               28204                 0 bps
     Low, non-TCP        :               28204                 0 bps
     Low, TCP            :                   0                 0 bps
     High, non-TCP       :                   0                 0 bps
     High, TCP           :                   0                 0 bps

[edit]
lab@Nuts-re0# run show interfaces queue lsq-1/1/0.0 forwarding-class DATA
May 01 15:54:37
  Logical interface lsq-1/1/0.0 (Index 92) (SNMP ifIndex 608)
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Burst size: 0
Queue: 0, Forwarding classes: DATA
  Queued:
    Packets              :                3155                64 pps
    Bytes                :             2448598            376544 bps
  Transmitted:
    Packets              :                3123                64 pps
    Bytes                :             2420394            376544 bps
    Tail-dropped packets :                   0                 0 pps
    RED-dropped packets  :                  32                 0 pps
     Low, non-TCP        :                  32                 0 pps
     Low, TCP            :                   0                 0 pps
     High, non-TCP       :                   0                 0 pps
     High, TCP           :                   0                 0 pps
    RED-dropped bytes    :               28204                 0 bps
     Low, non-TCP        :               28204                 0 bps
     Low, TCP            :                   0                 0 bps
     High, non-TCP       :                   0                 0 bps
     High, TCP           :                   0                 0 bps

[edit]
lab@Nuts-re0# run show interfaces queue lsq-1/1/0.0 forwarding-class DATA
May 01 15:54:38
  Logical interface lsq-1/1/0.0 (Index 92) (SNMP ifIndex 608)
Forwarding classes: 16 supported, 4 in use
```

```
Egress queues: 8 supported, 4 in use
Burst size: 0
Queue: 0, Forwarding classes: DATA
  Queued:
    Packets              :                3218                 63 pps
    Bytes                :             2500682             416672 bps
  Transmitted:
    Packets              :                3186                 63 pps
    Bytes                :             2472478             416672 bps
    Tail-dropped packets :                   0                  0 pps
    RED-dropped packets  :                  32                  0 pps
     Low, non-TCP        :                32                  0 pps
     Low, TCP            :               0                  0 pps
     High, non-TCP       :                0                  0 pps
     High, TCP           :               0                  0 pps
    RED-dropped bytes    :               28204                  0 bps
     Low, non-TCP        :             28204                  0 bps
     Low, TCP            :               0                  0 bps
     High, non-TCP       :                0                  0 bps
     High, TCP           :               0                  0 bps

[edit]
lab@Nuts-re0# run show interfaces queue lsq-1/1/0.0 forwarding-class DATA
May 01 15:54:39
  Logical interface lsq-1/1/0.0 (Index 92) (SNMP ifIndex 608)
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Burst size: 0
Queue: 0, Forwarding classes: DATA
  Queued:
    Packets              :                3406                188 pps
    Bytes                :             2646187            1164040 bps
  Transmitted:
    Packets              :                3366                180 pps
    Bytes                :             2612278            1118400 bps
    Tail-dropped packets :                   0                  0 pps
    RED-dropped packets  :                  38                  6 pps
     Low, non-TCP        :                38                  6 pps
     Low, TCP            :               0                  0 pps
     High, non-TCP       :                0                  0 pps
     High, TCP           :               0                  0 pps
    RED-dropped bytes    :               32162              31664 bps
     Low, non-TCP        :             32162              31664 bps
     Low, TCP            :               0                  0 bps
     High, non-TCP       :                0                  0 bps
     High, TCP           :               0                  0 bps

[edit]
lab@Nuts-re0# run show interfaces queue lsq-1/1/0.0 forwarding-class DATA
May 01 15:54:39
  Logical interface lsq-1/1/0.0 (Index 92) (SNMP ifIndex 608)
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Burst size: 0
```

```
Queue: 0, Forwarding classes: DATA
  Queued:
    Packets              :                  3406                   188 pps
    Bytes                :               2646187               1164040 bps
  Transmitted:
    Packets              :                  3366                   180 pps
    Bytes                :               2612278               1118400 bps
    Tail-dropped packets :                     0                     0 pps
    RED-dropped packets  :                    38                     6 pps
     Low, non-TCP        :                    38                     6 pps
     Low, TCP            :                     0                     0 pps
     High, non-TCP       :                     0                     0 pps
     High, TCP           :                     0                     0 pps
    RED-dropped bytes    :                 32162                 31664 bps
     Low, non-TCP        :                 32162                 31664 bps
     Low, TCP            :                     0                     0 bps
     High, non-TCP       :                     0                     0 bps
     High, TCP           :                     0                     0 bps

[edit]
lab@Nuts-re0# run show interfaces queue lsq-1/1/0.0 forwarding-class DATA
May 01 15:54:40
  Logical interface lsq-1/1/0.0 (Index 92) (SNMP ifIndex 608)
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Burst size: 0
Queue: 0, Forwarding classes: DATA
  Queued:
    Packets              :                  3469                    63 pps
    Bytes                :               2697746                412472 bps
  Transmitted:
    Packets              :                  3431                    65 pps
    Bytes                :               2665584                426448 bps
    Tail-dropped packets :                     0                     0 pps
    RED-dropped packets  :                    38                     0 pps
     Low, non-TCP        :                    38                     0 pps
     Low, TCP            :                     0                     0 pps
     High, non-TCP       :                     0                     0 pps
     High, TCP           :                     0                     0 pps
    RED-dropped bytes    :                 32162                     0 bps
     Low, non-TCP        :                 32162                     0 bps
     Low, TCP            :                     0                     0 bps
     High, non-TCP       :                     0                     0 bps
     High, TCP           :                     0                     0 bps
```

So with the q-pic-large buffer configuration added, the RED drops are still incrementing. Next, let's increase the buffer for the scheduler assigned to queue 0 to see if that helps alleviate the problem – let's change it from 70% to 85%.

```
[edit]
lab@Nuts-re0# show class-of-service schedulers
May 01 15:57:03
DATA-BE {
    transmit-rate percent 70;
    buffer-size percent 70;
    priority low;
}
VOICE-EF {
    transmit-rate percent 20;
    buffer-size percent 20;
    priority high;
}
APPS-AF {
    transmit-rate percent 5;
    buffer-size percent 5;
    priority medium-low;
}
NC {
    transmit-rate percent 5;
    buffer-size percent 5;
    priority strict-high;
}

[edit]
lab@Nuts-re0# set class-of-service schedulers DATA-BE buffer-size percent 85

[edit]
lab@Nuts-re0# set class-of-service schedulers VOICE-EF buffer-size percent 5

[edit]
lab@Nuts-re0# show class-of-service schedulers
May 01 15:57:22
DATA-BE {
    transmit-rate percent 70;
    buffer-size percent 85;
    priority low;
}
VOICE-EF {
    transmit-rate percent 20;
    buffer-size percent 5;
    priority high;
}
APPS-AF {
    transmit-rate percent 5;
    buffer-size percent 5;
    priority medium-low;
}
NC {
    transmit-rate percent 5;
    buffer-size percent 5;
    priority strict-high;
}

[edit]
lab@Nuts-re0# commit
May 01 15:57:30
commit complete
```

```
[edit]
lab@Nuts-re0# run clear interfaces statistics all
```

After the new configuration was committed and interface statistics cleared, the RED dropped packets counter is still incrementally increasing.

```
[edit]
lab@Nuts-re0# run show interfaces queue lsq-1/1/0.0 forwarding-class DATA
May 01 15:58:27
  Logical interface lsq-1/1/0.0 (Index 92) (SNMP ifIndex 608)
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Burst size: 0
Queue: 0, Forwarding classes: DATA
  Queued:
    Packets             :              3470                63 pps
    Bytes               :           2712445            390544 bps
  Transmitted:
    Packets             :              3452                63 pps
    Bytes               :           2698906            390544 bps
    Tail-dropped packets :                0                 0 pps
    RED-dropped packets  :               18                 0 pps
     Low, non-TCP        :              18                0 pps
     Low, TCP            :              0               0 pps
     High, non-TCP       :               0                0 pps
     High, TCP           :              0               0 pps
    RED-dropped bytes    :            13539                 0 bps
     Low, non-TCP        :           13539                0 bps
     Low, TCP            :              0               0 bps
     High, non-TCP       :               0                0 bps
     High, TCP           :              0               0 bps

[edit]
lab@Nuts-re0# run show interfaces queue lsq-1/1/0.0 forwarding-class DATA
May 01 15:58:30
  Logical interface lsq-1/1/0.0 (Index 92) (SNMP ifIndex 608)
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Burst size: 0
Queue: 0, Forwarding classes: DATA
  Queued:
    Packets             :              3785                63 pps
    Bytes               :           2956541            372296 bps
  Transmitted:
    Packets             :              3763                63 pps
    Bytes               :           2938625            372296 bps
    Tail-dropped packets :                0                 0 pps
    RED-dropped packets  :               22                 0 pps
     Low, non-TCP        :              22                0 pps
     Low, TCP            :              0               0 pps
     High, non-TCP       :               0                0 pps
     High, TCP           :              0               0 pps
    RED-dropped bytes    :              17916                   0 bps
     Low, non-TCP       :              17916                   0 bps
     Low, TCP          :                  0                   0 bps
     High, non-TCP      :                 0                   0 bps
     High, TCP         :                  0                   0 bps

[edit]
lab@Nuts-re0# run show interfaces queue lsq-1/1/0.0 forwarding-class DATA
May 01 15:58:39
  Logical interface lsq-1/1/0.0 (Index 92) (SNMP ifIndex 608)
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Burst size: 0
Queue: 0, Forwarding classes: DATA
  Queued:
    Packets             :               4479                 189 pps
    Bytes               :            3493525             1204800 bps
  Transmitted:
    Packets             :               4449                 181 pps
    Bytes               :            3467553             1140352 bps
    Tail-dropped packets :                 0                   0 pps
    RED-dropped packets  :                30                   8 pps
     Low, non-TCP       :                30                   8 pps
     Low, TCP          :                  0                   0 pps
     High, non-TCP      :                 0                   0 pps
     High, TCP         :                  0                   0 pps
    RED-dropped bytes    :              25972               64448 bps
     Low, non-TCP       :              25972               64448 bps
     Low, TCP          :                  0                   0 bps
     High, non-TCP      :                 0                   0 bps
     High, TCP         :                  0                   0 bps
```

So far, the q-pic-large-buffer and the buffer-size scheduler increase of queue 0 didn't eliminate the problem. They may have helped alleviate the problem, but they did not completely eliminate it. Another way to help with the issue is to add more bandwidth to the bundle, and to do this another T1 was provisioned to use the same LSQ bundle:

```
[edit]
lab@Nuts-re0# set interfaces t1-3/0/0:1:4 unit 0 family mlppp bundle lsq-1/1/0.0

[edit]
lab@Nuts-re0# show interfaces t1-3/0/0:1:4
May 01 16:00:09
unit 0 {
    family mlppp {
        bundle lsq-1/1/0.0;
    }
}

[edit]
lab@Nuts-re0# commit
May 01 16:00:11
commit complete
```

The change was also made to the other side of the circuit, on the M120.

```
 [edit]
lab@Reno-re0# set interfaces t1-3/0/0:1:4 unit 0 family mlppp bundle lsq-2/3/0.0

[edit]
lab@Reno-re0# show interfaces t1-3/0/0:1:4
unit 0 {
   family mlppp {
      bundle lsq-2/3/0.0;
   }
}

[edit]
lab@Reno-re0# commit
commit complete
```

After the configuration change is made, use the `show interfaces terse` command to check that the four T1 interfaces are assigned to the LSQ bundle as they should be:

```
[edit]
lab@Nuts-re0# run show interfaces terse | match "t1|lsq"
lsq-1/1/0             up    up
lsq-1/1/0.0           up    up   inet    10.1.1.1/30
t1-3/0/0:1:1          up    up
t1-3/0/0:1:1.0        up    up   mlppp    lsq-1/1/0.0
t1-3/0/0:1:2          up    up
t1-3/0/0:1:2.0        up    up   mlppp    lsq-1/1/0.0
t1-3/0/0:1:3          up    up
t1-3/0/0:1:3.0        up    up   mlppp    lsq-1/1/0.0
t1-3/0/0:1:4          up    up
t1-3/0/0:1:4.0        up    up   mlppp    lsq-1/1/0.0
```

Next, clear the interface statistics and verify that the packets are using the four member T1 links. You can see in the following snippet that the packets are close to evenly distributed across the four member links:

```
[edit]
lab@Nuts-re0# run clear interfaces statistics all

[edit]
lab@Nuts-re0# run show interfaces lsq-1/1/0 extensive
May 01 16:01:56
Physical interface: lsq-1/1/0, Enabled, Physical link is Up
  Interface index: 204, SNMP ifIndex: 603, Generation: 325
  Link-level type: LinkService, MTU: 1504
  Device flags   : Present Running
  Interface flags: Point-To-Point SNMP-Traps Internal: 0x4000
  Last flapped   : 2012-05-01 14:23:25 PDT (01:38:31 ago)
```

```
Statistics last cleared: 2012-05-01 16:01:22 PDT (00:00:34 ago)
Traffic statistics:
 Input  bytes  :              320                0 bps
 Output bytes  :          1949591           360776 bps
 Input  packets:                4                0 pps
 Output packets:             2534               62 pps
 IPv6 transit statistics:
  Input  bytes  :              0
  Output bytes  :              0
  Input  packets:              0
  Output packets:              0
Frame exceptions:
  Oversized frames             0
  Errored input frames          0
  Input on disabled link/bundle     0
  Output for disabled link/bundle   0
  Queuing drops                0
Buffering exceptions:
  Packet data buffer overflow      0
  Fragment data buffer overflow     0
Assembly exceptions:
  Fragment timeout             0
  Missing sequence number        0
  Out-of-order sequence number     0
  Out-of-range sequence number     0
Hardware errors (sticky):
  Data memory error            0
  Control memory error          0
Egress queues: 8 supported, 4 in use
Queue counters:      Queued packets  Transmitted packets     Dropped packets
  0 DATA                      0                 0                 0
  1 VOICE                     0                 0                 0
  2 APPS                      0                 0                 0
  3 NC                        0                 0                 0
Queue number:       Mapped forwarding classes
  0                DATA
  1                VOICE
  2                APPS
  3                NC

Logical interface lsq-1/1/0.0 (Index 92) (SNMP ifIndex 608) (Generation 299)
  Flags: Point-To-Point SNMP-Traps 0x4000 Encapsulation: Multilink-PPP
  Last flapped: 2012-05-01 15:52:54 PDT (00:09:02 ago)
  Bandwidth: 6144kbps
  Bundle links information:
    Active bundle links      4
    Removed bundle links     0
    Disabled bundle links     0
  Bundle options:
    MRRU                           1504
    Remote MRRU                     1504
    Drop timer period                500
    Inner PPP Protocol field compression enabled
    Sequence number format            long (24 bits)
    Fragmentation threshold          0
    Links needed to sustain bundle     1
    Multilink classes              0
    Link layer overhead            4.0 %
  Bundle status:
    Received sequence number      0x4b
    Transmit sequence number      0x9ec4
    Packet drops               0 (0 bytes)
    Fragment drops              0 (0 bytes)
    MRRU exceeded              0
    Fragment timeout             0
    Missing sequence number        0
    Out-of-order sequence number    0
    Out-of-range sequence number    0
    Packet data buffer overflow     0
    Fragment data buffer overflow   0
  Statistics        Frames       fps        Bytes         bps
  Bundle:
    Multilink:
      Input :            4        0         344          0
      Output:         2534        61      1972544       364120
    Network:
      Input :            4        0         320          0
      Output:         2534        62      1949591       360776
  Link:
    t1-3/0/0:1:1.0
      Up time: 00:09:02
      Input :            1        0          86          0
      Output:          617        16       494572        89176
    t1-3/0/0:1:2.0
      Up time: 00:09:02
      Input :            1        0          86          0
      Output:          625        16       491272        87520
    t1-3/0/0:1:3.0
      Up time: 00:09:02
      Input :            1        0          86          0
      Output:          645        14       491332        91736
    t1-3/0/0:1:4.0
      Up time: 00:01:14
      Input :            1        0          86          0
      Output:          647        15       495368        95688
  Multilink detail statistics:
  Bundle:
    Fragments:
      Input :            0        0           0          0
      Output:            0        0           0          0
    Non-fragments:
      Input :            4        0         344          0
      Output:         2584        63      2015709       401968
    LFI:
      Input :            0        0           0          0
      Output:            0        0           0          0
 NCP state: inet: Opened, inet6: Not-configured, iso: Not-configured, mpls: Not-
configured
```

```
    Protocol inet, MTU: 1500, Generation: 450, Route table: 0
      Flags: Sendbcast-pkt-to-re
      Addresses, Flags: Is-Preferred Is-Primary
        Destination: 10.1.1.0/30, Local: 10.1.1.1, Broadcast: Unspecified, Generation:
285
```

The interface statistics can now be checked to see if there are any more RED drops, and here the output confirms that there are no longer any drops:

```
[edit]
lab@Nuts-re0# run show interfaces queue lsq-1/1/0.0 forwarding-class DATA
May 01 16:02:30
  Logical interface lsq-1/1/0.0 (Index 92) (SNMP ifIndex 608)
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Burst size: 0
Queue: 0, Forwarding classes: DATA
  Queued:
    Packets              :                 5172                    63 pps
    Bytes                :              4004174                397160 bps
  Transmitted:
    Packets              :                 5172                    63 pps
    Bytes                :              4004174                397160 bps
    Tail-dropped packets :                    0                     0 pps
    RED-dropped packets  :                    0                     0 pps
     Low, non-TCP        :                    0                     0 pps
     Low, TCP            :                    0                     0 pps
     High, non-TCP       :                    0                     0 pps
     High, TCP           :                    0                     0 pps
    RED-dropped bytes    :                    0                     0 bps
     Low, non-TCP        :                    0                     0 bps
     Low, TCP            :                    0                     0 bps
     High, non-TCP       :                    0                     0 bps
     High, TCP           :                    0                     0 bps

[edit]
lab@Nuts-re0# run show interfaces queue lsq-1/1/0.0 forwarding-class DATA
May 01 16:02:32
  Logical interface lsq-1/1/0.0 (Index 92) (SNMP ifIndex 608)
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Burst size: 0
Queue: 0, Forwarding classes: DATA
  Queued:
    Packets              :                 5299                    63 pps
    Bytes                :              4104226                417016 bps
  Transmitted:
    Packets              :                 5299                    63 pps
    Bytes                :              4104226                417016 bps
    Tail-dropped packets :                    0                     0 pps
    RED-dropped packets  :                    0                     0 pps
     Low, non-TCP        :                    0                     0 pps
     Low, TCP            :                    0                     0 pps
     High, non-TCP       :                    0                     0 pps
     High, TCP          :                 0                 0 pps
    RED-dropped bytes    :                 0                 0 bps
     Low, non-TCP        :                 0                 0 bps
     Low, TCP           :                 0                 0 bps
     High, non-TCP       :                 0                 0 bps
     High, TCP          :                 0                 0 bps

[edit]
lab@Nuts-re0# run show interfaces queue lsq-1/1/0.0 forwarding-class DATA
May 01 16:02:32
  Logical interface lsq-1/1/0.0 (Index 92) (SNMP ifIndex 608)
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Burst size: 0
Queue: 0, Forwarding classes: DATA
  Queued:
    Packets             :              5299                63 pps
    Bytes              :           4104226            417016 bps
  Transmitted:
    Packets             :              5299                63 pps
    Bytes              :           4104226            417016 bps
    Tail-dropped packets :                 0                 0 pps
    RED-dropped packets  :                 0                 0 pps
     Low, non-TCP        :                 0                 0 pps
     Low, TCP           :                 0                 0 pps
     High, non-TCP       :                 0                 0 pps
     High, TCP          :                 0                 0 pps
    RED-dropped bytes    :                 0                 0 bps
     Low, non-TCP        :                 0                 0 bps
     Low, TCP           :                 0                 0 bps
     High, non-TCP       :                 0                 0 bps
     High, TCP          :                 0                 0 bps

[edit]
lab@Nuts-re0# run show interfaces queue lsq-1/1/0.0 forwarding-class DATA
May 01 16:02:33
  Logical interface lsq-1/1/0.0 (Index 92) (SNMP ifIndex 608)
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Burst size: 0
Queue: 0, Forwarding classes: DATA
  Queued:
    Packets             :              5362                63 pps
    Bytes              :           4159206            439840 bps
  Transmitted:
    Packets             :              5362                63 pps
    Bytes              :           4159206            439840 bps
    Tail-dropped packets :                 0                 0 pps
    RED-dropped packets  :                 0                 0 pps
     Low, non-TCP        :                 0                 0 pps
     Low, TCP           :                 0                 0 pps
     High, non-TCP       :                 0                 0 pps
     High, TCP          :                 0                 0 pps
    RED-dropped bytes    :                 0                 0 bps
```

```
     Low, non-TCP       :                 0                 0 bps
     Low, TCP           :                 0                 0 bps
     High, non-TCP      :                 0                 0 bps
     High, TCP          :                 0                 0 bps

[edit]
lab@Nuts-re0# run show interfaces queue lsq-1/1/0.0 forwarding-class DATA
May 01 16:02:34
  Logical interface lsq-1/1/0.0 (Index 92) (SNMP ifIndex 608)
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Burst size: 0
Queue: 0, Forwarding classes: DATA
  Queued:
    Packets              :              5425                63 pps
    Bytes                :           4205587            371048 bps
  Transmitted:
    Packets              :              5425                63 pps
    Bytes                :           4205587            371048 bps
    Tail-dropped packets :                 0                 0 pps
    RED-dropped packets  :                 0                 0 pps
     Low, non-TCP       :                 0                 0 pps
     Low, TCP           :                 0                 0 pps
     High, non-TCP      :                 0                 0 pps
     High, TCP          :                 0                 0 pps
    RED-dropped bytes    :                 0                 0 bps
     Low, non-TCP       :                 0                 0 bps
     Low, TCP           :                 0                 0 bps
     High, non-TCP      :                 0                 0 bps
     High, TCP          :                 0                 0 bps

[edit]
lab@Nuts-re0# run show interfaces queue lsq-1/1/0.0 forwarding-class DATA
May 01 16:02:35
  Logical interface lsq-1/1/0.0 (Index 92) (SNMP ifIndex 608)
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Burst size: 0
Queue: 0, Forwarding classes: DATA
  Queued:
    Packets              :              5488                63 pps
    Bytes                :           4248952            346920 bps
  Transmitted:
    Packets              :              5488                63 pps
    Bytes                :           4248952            346920 bps
    Tail-dropped packets :                 0                 0 pps
    RED-dropped packets  :                 0                 0 pps
     Low, non-TCP       :                 0                 0 pps
     Low, TCP           :                 0                 0 pps
     High, non-TCP      :                 0                 0 pps
     High, TCP          :                 0                 0 pps
    RED-dropped bytes    :                 0                 0 bps
     Low, non-TCP       :                 0                 0 bps
     Low, TCP           :                 0                 0 bps
     High, non-TCP      :                 0                 0 bps
     High, TCP          :                 0                 0 bps
```

```
[edit]
lab@Nuts-re0# run show interfaces queue lsq-1/1/0.0 forwarding-class DATA
May 01 16:02:35
  Logical interface lsq-1/1/0.0 (Index 92) (SNMP ifIndex 608)
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Burst size: 0
Queue: 0, Forwarding classes: DATA
  Queued:
    Packets            :             5488                   63 pps
    Bytes              :          4248952               346920 bps
  Transmitted:
    Packets            :             5488                   63 pps
    Bytes              :          4248952               346920 bps
    Tail-dropped packets :                0                    0 pps
    RED-dropped packets  :                0                    0 pps
     Low, non-TCP       :               0                    0 pps
     Low, TCP           :               0                    0 pps
     High, non-TCP      :               0                    0 pps
     High, TCP          :               0                    0 pps
    RED-dropped bytes    :               0                    0 bps
     Low, non-TCP       :               0                    0 bps
     Low, TCP           :               0                    0 bps
     High, non-TCP      :               0                    0 bps
     High, TCP          :               0                    0 bps

[edit]
lab@Nuts-re0# run show interfaces queue lsq-1/1/0.0 forwarding-class DATA
May 01 16:02:36
  Logical interface lsq-1/1/0.0 (Index 92) (SNMP ifIndex 608)
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Burst size: 0
Queue: 0, Forwarding classes: DATA
  Queued:
    Packets            :             5552                   64 pps
    Bytes              :          4298562               396880 bps
  Transmitted:
    Packets            :             5552                   64 pps
    Bytes              :          4298562               396880 bps
    Tail-dropped packets :                0                    0 pps
    RED-dropped packets  :                0                    0 pps
     Low, non-TCP       :               0                    0 pps
     Low, TCP           :               0                    0 pps
     High, non-TCP      :               0                    0 pps
     High, TCP          :               0                    0 pps
    RED-dropped bytes    :               0                    0 bps
     Low, non-TCP       :               0                    0 bps
     Low, TCP           :               0                    0 bps
     High, non-TCP      :               0                    0 bps
     High, TCP          :               0                    0 bps
```

```
[edit]
lab@Nuts-re0# run show interfaces queue lsq-1/1/0.0 forwarding-class DATA
May 01 16:02:36
  Logical interface lsq-1/1/0.0 (Index 92) (SNMP ifIndex 608)
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Burst size: 0
Queue: 0, Forwarding classes: DATA
  Queued:
    Packets              :                  5615                    63 pps
    Bytes                :               4355691                457032 bps
  Transmitted:
    Packets              :                  5615                    63 pps
    Bytes                :               4355691                457032 bps
    Tail-dropped packets :                     0                     0 pps
    RED-dropped packets  :                     0                     0 pps
     Low, non-TCP        :                     0                     0 pps
     Low, TCP            :                     0                     0 pps
     High, non-TCP       :                     0                     0 pps
     High, TCP           :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
     Low, non-TCP        :                     0                     0 bps
     Low, TCP            :                     0                     0 bps
     High, non-TCP       :                     0                     0 bps
     High, TCP           :                     0                     0 bps

[edit]
lab@Nuts-re0# run show interfaces queue lsq-1/1/0.0 forwarding-class DATA
May 01 16:02:37
  Logical interface lsq-1/1/0.0 (Index 92) (SNMP ifIndex 608)
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Burst size: 0
Queue: 0, Forwarding classes: DATA
  Queued:
    Packets              :                  5615                    63 pps
    Bytes                :               4355691                457032 bps
  Transmitted:
    Packets              :                  5615                    63 pps
    Bytes                :               4355691                457032 bps
    Tail-dropped packets :                     0                     0 pps
    RED-dropped packets  :                     0                     0 pps
     Low, non-TCP        :                     0                     0 pps
     Low, TCP            :                     0                     0 pps
     High, non-TCP       :                     0                     0 pps
     High, TCP           :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
     Low, non-TCP        :                     0                     0 bps
     Low, TCP            :                     0                     0 bps
     High, non-TCP       :                     0                     0 bps
     High, TCP           :                     0                     0 bps
```

```
[edit]
lab@Nuts-re0# run show interfaces queue lsq-1/1/0.0 forwarding-class DATA
May 01 16:02:38
  Logical interface lsq-1/1/0.0 (Index 92) (SNMP ifIndex 608)
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Burst size: 0
Queue: 0, Forwarding classes: DATA
  Queued:
    Packets              :                  5803                   188 pps
    Bytes                :               4495215               1116192 bps
  Transmitted:
    Packets              :                  5803                   188 pps
    Bytes                :               4495215               1116192 bps
    Tail-dropped packets :                     0                     0 pps
    RED-dropped packets  :                     0                     0 pps
     Low, non-TCP        :                     0                     0 pps
     Low, TCP            :                     0                     0 pps
     High, non-TCP       :                     0                     0 pps
     High, TCP           :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
     Low, non-TCP        :                     0                     0 bps
     Low, TCP            :                     0                     0 bps
     High, non-TCP       :                     0                     0 bps
     High, TCP           :                     0                     0 bps

[edit]
lab@Nuts-re0# run show interfaces queue lsq-1/1/0.0 forwarding-class DATA
May 01 16:02:39
  Logical interface lsq-1/1/0.0 (Index 92) (SNMP ifIndex 608)
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Burst size: 0
Queue: 0, Forwarding classes: DATA
  Queued:
    Packets              :                  5866                    63 pps
    Bytes                :               4545131                399328 bps
  Transmitted:
    Packets              :                  5866                    63 pps
    Bytes                :               4545131                399328 bps
    Tail-dropped packets :                     0                     0 pps
    RED-dropped packets  :                     0                     0 pps
     Low, non-TCP        :                     0                     0 pps
     Low, TCP            :                     0                     0 pps
     High, non-TCP       :                     0                     0 pps
     High, TCP           :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
     Low, non-TCP        :                     0                     0 bps
     Low, TCP            :                     0                     0 bps
     High, non-TCP       :                     0                     0 bps
     High, TCP           :                     0                     0 bps
```

Summary

The LSQ interface reported RED drops even though it appeared that the traffic rate was well below the capacity of the LSQ bundle.

The first thing we attempted was to add the q-pic-large-buffer configuration knob to give the PIC more delay buffer. This is a software knob that can be configured to allow the IQ PIC to have more memory available for delay buffer.

MORE? More information about the q-pic-large-buffer configuration knob can be found in the Juniper techpubs documentation: http://www.juniper.net/techpubs/en_US/junos11.4/topics/task/configuration/chassis-iq-pics-large-delay-buffers-configuring.html .

The RED drops still incremented even when q-pic-large-buffer was configured, so as a second step a higher buffer size allocation for the queue 0 scheduler was configured. The queue 0 scheduler already had 70% allocated for the buffer-size and it was increased to 85% to see if that would make any difference. The RED drops were still persistant and in order to fix the issue, more bandwidth was added to the bundle. This was done by adding another T1 interface in the bundle, so there were four T4 member links in the bundle instead of three.

Recipe 4

Traffic in the Wrong Egress Queue

This recipe is intended to show you the troubleshooting steps and associated CLI commands that can be used to help identify a problem where traffic is being forwarded out of the wrong egress queue.

Problem

Packets are being sent out of the wrong egress queue. Certain traffic that is intended to have used the Expedited-Forwarding queue is not being forwarded out of that queue.

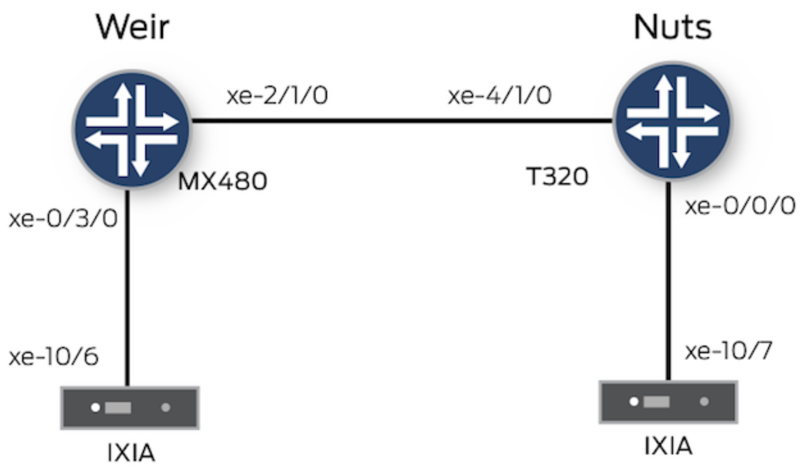
This type of problem could occur due to many reasons, including various configuration problems, a software issue, or because the packets are not correctly marked with the Class of Service ToS (Type of Service) bits. The most common cause is a configuration issue because there are so many factors that come into play from the configuration side. For example, an edge router might have a MF (Multi-Field) Classifier applied on the ingress interface to mark the ToS bits of a particular traffic stream, and the MF Classifier would need to be correctly configured to do this. Then, the egress interface might have a rewrite rule applied to it, which would need to be taken into consideration. A re-write rule would take the egress packets and re-write the ToS bits, so it is important to have the correct configuration and to be aware of this . Finally, one of the internal network routers could have a BA (Behavior Aggregate) Classifier configured to take those packets with the specific ToS markings to map the traffic to a specific egress queue. So there are several areas in the configuration that can influence the traffic to use a particular queue for forwarding.

Solution

When you observe that packets are arriving on a router with a certain DSCP (Differentiated Services Code Point) bit marking, and they are being forwarded out of the wrong egress interface queue, it's important to note this because the intended scheduler may not be used for this particular traffic and instead the router would use a different scheduler. The wrong egress queue could cause problems and might lead to dropped packets, depending on how the Class of Service is configured. For example, it wouldn't be a good idea to have delayed sensitive voice traffic use the same scheduler as lower priority data traffic – it might cause jitter or even dropped calls!

Now that you understand that there are several ways for a wrong egress queue problem to happen, let's investigate:

The lab routers Weir and Nuts are MX480 and T320 respectively, and both are running Junos 11.4R1.14 and they are connected to an IXIA Traffic Generator.



In this recipe's topology, the IXIA traffic generators are sending a total of about 150,000 frames per second in both directions. They are sending three streams of traffic:

- One stream is 50,000 frames per second, in both directions, and has a DSCP bit marking of best-effort, or "000000".

- The second stream is bi-directional at 50,000 frames per second and has a DSCP bit marking of assured-forwarding, or "001010".
- The final stream is also a bi-directional rate of 50,000 frames per second and has a DSCP bit marking of expedited-forwarding, or "101110".

The MX480 and T320 routers are both configured to classify the packets accordingly, based on the DSCP bit markings:

- Best-effort traffic should use queue 0
- Expedited-forwarding traffic should use queue 1
- Assured-forwarding traffic should use queue 2

Let's start with the T320 and the traffic arriving on the xe-0/0/0 interface from the IXIA traffic generator.:

```
{master}[edit]
lab@Nuts-re0# run show interfaces xe-0/0/0 extensive
Physical interface: xe-0/0/0, Enabled, Physical link is Up
<snip>
  Traffic statistics:
   Input  bytes  :          2492372020            949545544 bps
   Output bytes  :          2491626346            950490800 bps
   Input  packets:             3150057               150003 pps
   Output packets:             3150057               150003 pps
<snip>
  Ingress queues: 8 supported, 4 in use
  Queue counters:       Queued packets  Transmitted packets      Dropped packets
    0 BE                       1050019              1050019                    0
    1 EF                       1050019              1050019                    0
    2 AF                       1050019              1050019                    0
    3 NC                             0                    0                    0
  Egress queues: 8 supported, 4 in use
  Queue counters:       Queued packets  Transmitted packets      Dropped packets
    0 BE                       1100020              1100020                    0
    1 EF                       1100020              1100020                    0
    2 AF                       1100020              1100020                    0
    3 NC                             0                    0                    0
```

Notice that there are approximately 150,000 ingress packets and egress packets here, as you would expect. Also, the egress packets are using the three queues: best-effort (BE), expedited-forwarding (EF), and assured-forwarding (AF). So far, everything looks normal.

Now, let's look at the T320 xe-4/1/0 interface, which faces the other router, and you'll see that there are about 150,000 frames, both sent and received, which is correct. The egress statistics show that the BE, AF, and EF queues are being used as expected:

```
{master}[edit]
lab@Nuts-re0# run show interfaces xe-4/1/0 extensive
Physical interface: xe-4/1/0, Enabled, Physical link is Up
<snip>
  Traffic statistics:
   Input  bytes  :          3203020206           947914160 bps
   Output bytes  :          3204594535           948937432 bps
   Input  packets:             4050044              150003 pps
   Output packets:             4050043              150000 pps
<snip>
  Ingress queues: 8 supported, 4 in use
  Queue counters:       Queued packets  Transmitted packets      Dropped packets
    0 BE                       1350014              1350014                    0
    1 EF                       1350014              1350014                    0
    2 AF                       1350014              1350014                    0
    3 NC                             2                    2                    0
  Egress queues: 8 supported, 4 in use
  Queue counters:       Queued packets  Transmitted packets      Dropped packets
    0 BE                       1350013              1350013                    0
    1 EF                       1350014              1350014                    0
    2 AF                       1350013              1350013                    0
    3 NC                             3                    3                    0
```

The next output is from the MX480 and the xe-2/1/0 interface, which is facing the T320. This link also shows the bi-directional 150,000 frames per second, and, it correctly shows transmitted packets in the BE, AF, and EF queues. However, this link doesn't have Ingress queuing statistics because it does not have an IQ2 PIC like the one the T320 has.

```
{master}[edit]
lab@Weir-re0# run show interfaces xe-2/1/0 extensive
Physical interface: xe-2/1/0, Enabled, Physical link is Up
<snip>
  Traffic statistics:
   Input  bytes  :          6278164065           926958832 bps
   Output bytes  :          6277886854           927521264 bps
   Input  packets:             8121833              149999 pps
   Output packets:             8121834              150000 pps
<snip>
  Egress queues: 8 supported, 4 in use
  Queue counters:       Queued packets  Transmitted packets      Dropped packets
    0 BE                       2520023              2520024                    0
    1 EF                       2520024              2520024                    0
    2 AF                       2520024              2520024                    0
    3 NC                             6                    6                    0
```

Now, in the next capture, the MX480 xe-0/3/0 interface shows a bi-directional traffic rate of 150,000 frames per second. However, by looking at the egress queue statistics, traffic is only using the BE and AF queues. Here's the problem – our EF traffic stream is not using the EF queue as expected :

```
{master}[edit]
lab@Weir-re0# run show interfaces xe-0/3/0 extensive
Physical interface: xe-0/3/0, Enabled, Physical link is Up
<snip>
  Traffic statistics:
   Input  bytes  :          6916099703            927909264 bps
   Output bytes  :          6916303021            927421488 bps
   Input  packets:             8947830               150000 pps
   Output packets:             8947833               150000 pps
<snip>
  Egress queues: 8 supported, 4 in use
  Queue counters:       Queued packets  Transmitted packets      Dropped packets
    0 BE                       3063285              3063285                    0
    1 EF                             0                    0                    0
    2 AF                       3063285              3063285                    0
    3 NC                             0                    0                    0
```

You can see that the expedited-forwarding queue is transmitting no packets, which of course could lead to problems such as the assured-forwarding queue being oversubscribed, and the EF-marked traffic not having the intended class of service scheduler configuration.

The `class-of-service` configurations on the MX480 and T320 are identical, so let's look at the configuration on the MX480 router , the one with the problem, to see if the classifier is applied to both interfaces:

```
{master}[edit]
lab@Weir-re0# show class-of-service
classifiers {
    dscp CLASSIFIER_1 {
        forwarding-class BE {
            loss-priority low code-points 000000;
        }
        forwarding-class AF {
            loss-priority low code-points 001010;
        }
        forwarding-class EF {
            loss-priority high code-points 101110;
        }
        forwarding-class NC {
            loss-priority high code-points 110000;
        }
    }
}
forwarding-classes {
    queue 0 BE;
    queue 1 EF;
    queue 2 AF;
    queue 3 NC;
}
interfaces {
    xe-0/3/0 {
        scheduler-map CORE;
```

```
            unit 0 {
                classifiers {
                    dscp CLASSIFIER_1;
                }
            }
        }
        xe-2/1/0 {
            scheduler-map CORE;
            unit 0 {
                classifiers {
                    dscp CLASSIFIER_1;
                }
            }
        }
    }
    scheduler-maps {
        CORE {
            forwarding-class BE scheduler CORE-BE;
            forwarding-class AF scheduler CORE-AF;
            forwarding-class EF scheduler CORE-EF;
            forwarding-class NC scheduler CORE-NC;
        }
    }
    schedulers {
        CORE-BE {
            transmit-rate remainder;
            buffer-size {
                remainder;
            }
            priority low;
        }
        CORE-AF {
            transmit-rate percent 10;
            buffer-size percent 10;
            priority medium-high;
        }
        CORE-EF {
            transmit-rate percent 25;
            buffer-size percent 25;
            priority high;
        }
        CORE-NC {
            transmit-rate percent 5;
            buffer-size percent 5;
            priority strict-high;
        }
    }
```

There are no problems with the configuration. Notice that the classifier, `CLASSIFIER_1`, is correctly configured because it maps the DSCP EF code-points `101110` to the EF forwarding-class. Also, this classifier is applied to the ingress interface xe-2/1/0, so any packets with the DSCP bits `101110`, which are EF, should be assigned to the EF queue. Let's keep investigating.

The `show class-of-service interface <interface-name>` command should show which scheduler-map, classifiers, or re-write rules are applied to a particular interface. The CLI command is run using the ingress xe-2/1/0 interface shown below:

```
{master}[edit]
lab@Weir-re0# run show class-of-service interface xe-2/1/0
Physical interface: xe-2/1/0, Index: 145
Queues supported: 8, Queues in use: 4
  Scheduler map: CORE, Index: 46629
  Congestion-notification: Disabled

  Logical interface: xe-2/1/0.0, Index: 352
    Object                 Name                  Type                     Index
    Classifier             CLASSIFIER_1          dscp                     42245
```

The custom classifier named CLASSIFIER_1 is correctly applied to the xe-2/1/0 interface in the output. This confirms that if traffic is arriving on the MX480 xe-2/1/0 interface with the EF DSCP bit value of 101110, then it should be sent out of the xe-0/3/0 interface in the EF queue.

Let's use the `show class-of-service classifier name <CLASSIFIER_1>` command to show the specific DSCP code point bit values and associated forwarding-class for those values to make sure that they are correctly assigned:

```
{master}[edit]
lab@Weir-re0# run show class-of-service classifier name CLASSIFIER_1
Classifier: CLASSIFIER_1, Code point type: dscp, Index: 42245
  Code point         Forwarding class                    Loss priority
  000000             BE                                  low
  001010             AF                                  low
  101110             EF                                  high
  110000             NC                                  high
```

The classifier output looks normal. Notice that the correct code points are assigned to the appropriate forwarding class.

Next, let's check to see if firewall filters are applied to the interfaces on the MX480, which could be causing this problem. In Junos, firewall filters are used as a multi-field (MF) classifier. The MF classifier takes precedence over a behavior-aggregate (BA) classifier defined at the [edit class-of-service classifier] hierarchy.

```
{master}[edit]
lab@Weir-re0# show interfaces xe-2/1/0
unit 0 {
    family inet {
        address 10.1.1.1/30;
    }
}

{master}[edit]
lab@Weir-re0# show interfaces xe-0/3/0
unit 0 {
    family inet {
        address 192.168.1.1/30;
    }
}
```

As shown here, there are no firewall filters applied to the interfaces, so the router should use the BA classifier, named CLASSIFIER_1, that we have configured.

A good troubleshooting practice is to confirm that the packets arriving on the MX480 xe-2/1/0 ingress interface are marked correctly with the intended DSCP bit values. In order to do that, you make sure that the DSCP bit patterns being used are mapped to the correct code-point alias. The `show class-of-service code-point-aliases dscp` command is used to show us this for DSCP (the ones being used are highlighted):

```
{master}[edit]
lab@Weir-re0# run show class-of-service code-point-aliases dscp
Code point type: dscp
  Alias                 Bit pattern
  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
  cs6                   110000
  cs7                   111000
  ef                    101110
  nc1                   110000
  nc2                   111000
```

You see that the aliases are *af11*, *be*, and *ef*. In order to confirm that the packets are arriving on the MX480 xe-2/1/0 interface, you can apply an ingress firewall filter counter on that link to *count* the packets for the associated DSCP bit marking.

ALERT! Add the default `accept` term since default behavior for firewall filters in Junos is to discard all packets not matching any of the preceding terms.

```
{master}[edit]
lab@Weir-re0# show firewall
family inet {
    filter TEST_DSCP {
        interface-specific;
        term 1 {
            from {
                dscp be;
            }
            then {
                count dscp-be;
                accept;
            }
        }
        term 2 {
            from {
                dscp af11;
            }
            then {
                count dscp-af11;
                accept;
            }
        }
        term 3 {
            from {
                dscp ef;
            }
            then {
                count dscp-ef;
                accept;
            }
        }
        term default {
            then accept;
        }
    }
}
```

Let's apply this firewall filter on the xe-2/1/0 interface as an ingress filter so it will count packets coming into the router:

```
{master}[edit]
lab@Weir-re0# show interfaces xe-2/1/0
unit 0 {
    family inet {
        filter {
            input TEST_DSCP;
        }
        address 10.1.1.1/30;
    }
}
```

Next, let's check this filter and see that the BE, AF1 1, and EF counters are incrementally increasing.

```
{master}[edit]
lab@Weir-re0# run show firewall

Filter: TEST_DSCP-xe-2/1/0.0-i
Counters:
Name                                                Bytes              Packets
dscp-af11-xe-2/1/0.0-i                          110949044               143510
dscp-be-xe-2/1/0.0-i                            111186538               143510
dscp-ef-xe-2/1/0.0-i                            111189832               143510

{master}[edit]
lab@Weir-re0# run show firewall

Filter: TEST_DSCP-xe-2/1/0.0-i
Counters:
Name                                                Bytes              Packets
dscp-af11-xe-2/1/0.0-i                          525384419               679759
dscp-be-xe-2/1/0.0-i                            525570125               679759
dscp-ef-xe-2/1/0.0-i                            526456377               679758
```

You can see that the packets arriving on the MX480 router on this xe-2/1/0 interface are indeed marked correctly with the DSCP bit markings, indicating that the problem is somewhere on the MX480 router – the BA classifier is not working correctly.

You already know that the BA classifier, named CLASSIFIER_1, is applied to the xe-2/1/0 interface as it should be. So let's confirm if the classifier is correctly applied to the linecard that owns the xe-2/1/0 port, which is FPC2. In order to do this, you must first know the CLASSIFIER_1 index number. The index number is just a random numerical value that the system assigns each classifier. You can get the Classifier index number using the `show class-of-service interface <interface-name>` command:

```
{master}[edit]
lab@Weir-re0# run show class-of-service interface xe-2/1/0
Physical interface: xe-2/1/0, Index: 145
Queues supported: 8, Queues in use: 4
  Scheduler map: CORE, Index: 46629
  Congestion-notification: Disabled

  Logical interface: xe-2/1/0.0, Index: 352
    Object                  Name                  Type                    Index
    Classifier              CLASSIFIER_1          dscp                    42245
```

And the CLASSIFIER_1 classifier has an index value of 42245.

Now let's log in to the PFE shell, the FPC2 linecard, in order to confirm that the classifier is properly programmed on the xe-2/1/0 interface.

WARNING! Do not run shell commands in production without JTAC supervision. Such commands may potentially impact traffic.

This book is about using all the tools at your disposal. You now know that using shell commands is not supported or endorsed, but let's proceed like careful, prudent engineers who want to solve a problem.

Let's log in to FPC2:

```
{master}[edit]
lab@Weir-re0# run start shell pfe network fpc2

ADPC platform (1200Mhz MPC 8548 processor, 1024MB memory, 512KB flash)
ADPC2(Weir-re0 vty)#
```

The CLASSIFIER_1 index value is 42245 so let's get the xe-2/1/0.0 index value. The system creates a unique index number for each physical interface (IFD) and each logical interface (IFL), so use the PFE shell command `show ifl brief` to view the IFL index mappings to interface name.

```
ADPC2(Weir-re0 vty)# show ifl brief
Index  Name                  Type           Encapsulation   Flags
-----  -------------------- -------------  --------------  ------
  <snip>
  351  xe-0/3/0.0            Ethernet       Ethernet        0x000000000400c000
  352  xe-2/1/0.0            Ethernet       Ethernet        0x000000000400c000
```

You can see the xe-2/1/0.0 IFL maps to index number 352. Another way to get the IFL index number for interface xe-2/1/0.0 is to check the previously used `show class-of-service interface xe-2/1/0` command output – it shows that this logical interface xe-2/1/0.0 has an index number of 352 assigned to it. The more you learn about Junos, the more you notice that there is often more than one way to accomplish certain tasks.

Now let's make sure CLASSIFIER_1, index 42245, is programmed on xe-2/1/0.0, IFL index 352. To view the class-of-service programming on a particular IFL, use the PFE shell command `show cos ifl-entry <ifl-index-number>`:

```
ADPC2(Weir-re0 vty)# show cos ifl-entry 352
Cos-ifl entry for ifl xe-2/1/0.0 (352)
==================================
Ifl-index                  : 352
Ifd-index                  : 0
cos flags                  : 0x0
```

```
Class-slot                    : 0x8
Class-slot table              : 0x0
rw_flags[0]                   : 0x0
rw_flags[1]                   : 0x0
rw_flags[2]                   : 0x0
rw_table_id                   : 0
queue_num                     : 0
classifiers[      unknown] : 0
classifiers[         DSCP] : 0
classifiers[          EXP] : 0
classifiers[   IEEE 802.1] : 0
classifiers[IP precedence] : 0
<snip>
```

Notice in the output that the DSCP classifiers assigned to xe-2/1/0.0, IFL 352, are "0", meaning that no classifiers are programmed on that interface on the FPC2 linecard! From the routing engine's view, the classifier CLASSIFIER_1 was correctly applied to xe-2/1/0.0, as seen previously with the `show class-of-service interface xe-2/1/0` command. However, as just witnessed, from the FPC2 linecard's point of view there is no DSCP classifier applied on the ingress xe-2/1/0.0 interface! You would expect the classifier named CLASSIFIER_1 to be applied to it. So this is the problem and it explains why the packets are not being classified correctly, and thus not being forwarded out of the correct egress queue on xe-0/3/0.

When the CLASSIFIER_1 configuration is first committed in the configuration, it is stored on the routing engine. The routing engine should propagate the CLASSIFIER_1 configuration to the appropriate linecards, such as FPC2, in our case, since this is where the interface xe-2/1/0 resides. It would then be programmed on FPC2. So this looks like a problem because the CLASSIFIER_1 configuration, for some reason, did not get propagated to the FPC2 linecard, or had a problem getting programmed on the FPC2 linecard.

Another way to see if the classifier is programmed correctly is to use the `show cos classifier` PFE shell command. It shows the classifier id as the index number and on which IFL's it is programmed.

```
ADPC2(Weir-re0 vty)# show cos classifier
    Classifier id:      9 num-of-entries: 64 type: DSCP IPv6
       FEB download status: none
    Classifier id:     10 num-of-entries:  8 type: EXP
       FEB download status:
         FEB 2, chip 1, phy_index:0x764132ab, ifl: 352

    Classifier id:     13 num-of-entries:  8 type: IP precedence
       FEB download status: none
    Classifier id:  42245 num-of-entries:  4 type: DSCP
       FEB download status: none
```

The output shows that our classifier index 42245 is not downloaded to any IFLs, and it is expected it to be downloaded and programmed to the IFL 352, which is xe-2/1/0.0. So the shell command shows where the problem is.

To make sure that the classifier programming configuration on the linecard is the same as the one configured on the routing engine (at the [edit class-of-service classifier] level), the PFE shell `show cos classifier 42245` command can be used. The CP column in the following output shows the code-points value for DSCP in hexadecimal, and those code points can be converted to binary to confirm that the code-point value matches the one we have defined in the router configuration.. The "FC" is the forwarding-class.

```
ADPC2(Weir-re0 vty)# show cos classifier 42245
    classifier id:  42245    type: DSCP    slot(chip): -----
    cos flags    : 0x0    cl flags: 0x0
    CP  FC  PLP     CP  FC  PLP     CP  FC  PLP     CP  FC  PLP
   ------------    ------------    ------------    ------------
   0x00  0  0      0x0a  2  0      0x2e  1  1      0x30  3  1
```

You can see that for FC 0, the CP is 0x00, which in binary is still "000000". This matches the best-effort configuration that we have. For FC 1, the CP is 0x2e, which in binary is "101110" and this matches the expedited-forwarding value correctly. Next look at FC 2, and the CP is 0x0a, and that is "001010" in binary, which maps correctly to the assured-forwarding.

This sequence shows us that the classifier configuration for the FPC2 linecard itself matches the configuration at the [edit class-of-service classifier] hierarchy. However, the problem is that this classifier programming in FPC2 is not applied to the xe-2/1/0.0 interface as it should be.

There are a few different ways you can try to fix this problem, including removing and re-adding the classifier configuration, restarting the cosd daemon, physically bouncing the xe-2/1/0 port, restarting the FPC2 card, or completing an RE switchover/reboot.

Let's remove the class-of-service classifier and then re-add it to the configuration again. The following configuration changes were completed to do this:

```
{master}[edit]
lab@Weir-re0# deactivate class-of-service interfaces xe-2/1/0 unit 0 classifiers

{master}[edit]
lab@Weir-re0# commit
re0:
configuration check succeeds
re1:
```

```
commit complete
re0:
commit complete

{master}[edit]
lab@Weir-re0# activate class-of-service interfaces xe-2/1/0 unit 0 classifiers

{master}[edit]
lab@Weir-re0# commit
re0:
configuration check succeeds
re1:
commit complete
re0:
commit complete
```

At the same time that these changes were made, another window was open and running a capture from the shell, and using the `rtsockmon -t` shell command, it will show the add, delete, or change requests between the routing-engine and the packet forwarding engine (PFE). When the classifier configuration is removed from xe-2/1/0.0, the delete request is captured and when the classifier is added back, the add request is seen going to the PFE:

```
{master}
lab@Weir-re0> start shell user root
Password:
root@Weir-re0%
root@Weir-re0% rtsockmon -t
            sender   flag   type      op
[09:57:13] cosd    P   gencfg    delete  mtype=COS BLOB seq 1 kid=0 minor
type=<classifier to ifl>: ifl = 352, table type = DSCP, table id = 42245, cos_flags 0x0
<snip>
[09:57:25] cosd    P   gencfg    add    mtype=COS BLOB seq 9 kid=0 minor
type=<classifier to ifl>: ifl = 352, table type = DSCP, table id = 42245, cos_flags 0x0
```

Notice that the add request correctly includes the classifier index 42245 mapped to IFL 352.

Next, let's again go to the PFE shell, on the FPC2 linecard, to confirm that it is programmed correctly.

WARNING! Running commands on the linecards has the potential to cause impact, they should first be run in the lab and with JTAC instructions.

```
{master}[edit]
lab@Weir-re0# run start shell pfe network fpc2

ADPC platform (1200Mhz MPC 8548 processor, 1024MB memory, 512KB flash)

ADPC2(Weir-re0 vty)# show cos ifl-entry 352
Cos-ifl entry for ifl xe-2/1/0.0 (352)
==================================
Ifl-index                  : 352
Ifd-index                  : 0
cos flags                  : 0x0
Class-slot                 : 0x8
Class-slot table           : 0x0
rw_flags[0]                : 0x0
rw_flags[1]                : 0x0
rw_flags[2]                : 0x0
rw_table_id                : 0
queue_num                  : 0
classifiers[      unknown] : 0
classifiers[         DSCP] : 42245
classifiers[          EXP] : 0
classifiers[   IEEE 802.1] : 0
classifiers[IP precedence] : 0
<snip>
```

You can see that the classifier is correctly applied to xe-2/1/0.0. Notice that the IFL 352 index, xe-2/1/0.0, correctly has the CLASSIFIER_1 classifier index 42245 programmed.

The packets should now be leaving the EF queue from this MX480 as expected:

```
{master}[edit]
lab@Weir-re0# run show interfaces xe-0/3/0 extensive
Physical interface: xe-0/3/0, Enabled, Physical link is Up
<snip>
  Traffic statistics:
   Input  bytes  :        418225122661           926578344 bps
   Output bytes  :        418220495879           927334528 bps
   Input  packets:           541045176              149998 pps
   Output packets:           541045174              149997 pps
<snip>
  Egress queues: 8 supported, 4 in use
  Queue counters:       Queued packets  Transmitted packets      Dropped packets
    0 BE                     181476340            181476340                    0
    1 EF                      12942203             12942203                    0
    2 AF                     179823330            179823330                    0
    3 NC                             0                    0                  0
```

Summary

This recipe witnessed packets being sent out of an MX480 router that were not leaving in the correct egress queue. This type of problem could impact traffic since the desired scheduler configuration would not be used and instead the traffic in the wrong queue would be subject to another queue's scheduler map. These types of problems are very broad and can have many causes. Several troubleshooting steps were shown to help you narrow down the solution to the problem.

First, the problem was noticed with the `show interfaces extensive <interface-name>`CLI command. This CLI command shows the different egress queues and the amount of packets transmitted out of each queue. The problem was noticed here because the EF queue did not show any transmitted packets when we expected there to be packets sent out of this queue.

So the `show class-of-service interface <interface-name>`command was used on the ingress interface to make sure the classifier was applied to this interface. The classifier was correctly applied to the ingress interface as one would expect. Then the `show class-of-service classifier name <classifier-name>` command was run to make sure the classifier is correctly configured.

Then we made sure that there were no firewall filters applied to the ingress interface that might cause this problem. Firewall filters can be used as a MF classifier, so it could direct traffic to specific queues if configured to do so (in Junos MF classifiers take precedence over BA classifiers). In our case, there were no ingress firewall filters configured.

As a troubleshooting step, an ingress firewall filter was added to count the BE, AF, and EF traffic to verify that the packets arriving on the ingress interface have the correct DSCP bit markings. It was observed that the router received packets marked correctly with the EF bit markings, so at this point, the problem was isolated to the local router.

Next, we logged into the linecard associated with the ingress interface to verify that the BA Classifier was correctly assigned to the interface. We used the shell commands `show cos ifl-entry <ifl-index>` and `show cos classifier <classifier-index>` and noticed the problem here – the classifier was not programmed to the ingress interface on the linecard.

Finally, to fix the problem, we simply deactivated and activated the class of service classifier configuration from the ingress interface. This forced the classifier to be removed from the linecard and re-added again. After doing this, we observed the classifier programmed correctly on the linecard and also observed the traffic using the correct queue. Some other alternatives that could have fixed the problem are restarting the class-of-service daemon (cosd), restarting the linecard, or performing a routing engine mastership switch or reboot. We chose to deactivate and reactivate the classifier configuration from the ingress interface because this option caused the least impact on service. The other alternatives mentioned here may cause more impact on service, so please use them with caution.

Recipe 5

MF Classifier Not Operating Properly

This recipe shows the Junos commands necessary to troubleshoot problems related to a MF Classifier not working as intended.

Problem

A configured Multifield (MF) Classifier is applied to an interface, but is not operating properly as desired. Multifield Classifiers are usually applied on the edge devices in the network. They are firewall filters applied to an interface that matches on various attributes of packets and forwards the packets to different queues based on these attributes. Core routers usually include a Behavior Aggregate (BA) Classifier. So the edge routers tend to have the MF Classifier to classify the traffic in desired queues and rewrite any ToS bits, while the core routers have the BA Classifier to take the traffic and forward it to appropriate queues based on those ToS bits.

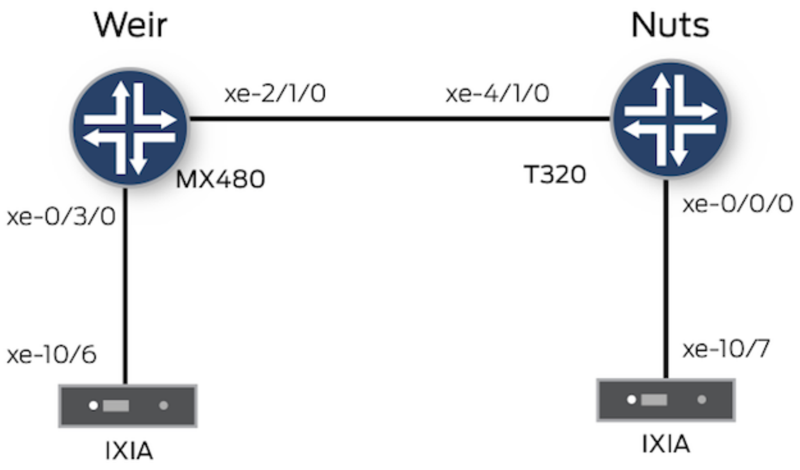
A problem with a MF Classifier can cause a range of different symptoms, depending on what the specific problem is. For example, if a particular traffic stream is intended to match on a certain MF Classifier filter term but instead goes to a different term, the traffic stream may leave the egress interface out of the wrong queue or there may be packet drops.

The most common cause of a MF Classifier not operating as intended is a configuration mistake. The larger the firewall filter acting as an MF Classifier, the more chances there are for mistakes to be made. It's always a good idea to test the filter on a lab router before applying it to a production router, in order to make sure specific traffic takes the desired term in the firewall filter. Counters can be included in the firewall filter terms to ensure that packets are hitting a desired term.

Solution

Many engineers configure a MF Classifier on edge devices that need to differentiate between different traffic streams based on IP header information such as addresses, protocols, or port numbers. They use a Junos MF Classifier which is a firewall filter that assigns the traffic to a particular forwarding-class.

In the lab, routers Weir and Nuts are MX480 and T320, respectively. They are both running Junos 11.4R1.14 and they are connected to an IXIA Traffic Generator.



In the lab's topology, the IXIA traffic generators are sending about 3,000 frames per second from IXIA port xe-10/7 to xe-10/6. The IXIA is sending three streams of traffic:

Stream #1

- Source IP address: 192.168.2.2
- Destination IP range: 100.100.100.1 through 100.100.100.100
- Protocol: udp
- Source port: 111
- Destination port: 111

- TOS bit markings: none
- Size: Random 64 through 1518
- Rate: 1,000 fps

Stream #2

- Source IP address: 192.168.2.2
- Destination IP address: 100.100.100.1
- Protocol: udp
- Source port: 9,000
- Destination port: 10,000
- TOS bit markings: none
- Size: Random 64 through 1518
- Rate: 1,000 fps

Stream #3

- Source IP address: 192.168.2.2
- Destination IP address: 100.100.100.200
- Protocol: ESP
- Source port: N/A
- Destination port: N/A
- TOS bit markings: none
- Size: Random 64 through 1518
- Rate: 1,000 fps

The T320 router has a Multifield (MF) Classifier to classify the incoming traffic from IXIA. This MF Classifier distinguishes the traffic based on the IP properties and assigns a particular forwarding class to each. MF Classifiers are configured as firewall filters. The T320 has the following MF Classifier filter configuration:

```
[edit]
lab@Nuts-re0# show firewall family inet filter MF-FILTER
term Application_1 {
    from {
        destination-address {
            100.100.100.1/32;
        }
```

```
            protocol [ udp tcp ];
            destination-port 10000;
        }
        then forwarding-class AF;
    }
    term Application_2 {
        from {
            destination-address {
                100.100.100.200/32;
            }
            protocol esp;
            port [ 50 51 500 ];
        }
        then forwarding-class EF;
    }
    term DEFAULT {
        then accept;
    }
```

The `Application_1` term in the filter is intended to match on UDP or TCP packets with a destination port of 10000 and destination IP address of 100.100.100.1/32. This term then sends the packets to forwarding-class AF and one would expect the IXIA *Stream* #2 traffic to hit this filter term and use the AF queue.

The `Application_2` term matches on ESP packets with a destination address of 100.100.100.200/32, and a port of 50, 51, or 500. These packets should use the EF forwarding-class and one would expect the IXIA *Stream #3* traffic to hit this term and use the EF queue.

The last term, `DEFAULT`, just accepts all traffic. Since the default behavior of a firewall filter in Junos is to discard all packets that do not match preceding terms, you need to include this term in order to accept the rest of the packets. IXIA is not sending traffic with any TOS bit markings, so the traffic that hits this `DEFAULT` term just uses the BE queue.

The filter then needs to be applied to an interface. Even though it is defined, it will not be evaluated until it is applied to an interface. In our topology, the interface facing IXIA is xe-0/0/0, so let's apply it as an inbound filter on that interface:

```
[edit]
lab@Nuts-re0# show interfaces xe-0/0/0
unit 0 {
    family inet {
        filter {
            input MF-FILTER;
        }
        address 192.168.1.5/30;
    }
}
```

Now let's verify that the packets are coming into xe-0/0/0 on the T320:

```
[edit]
lab@Nuts-re0# run show interfaces xe-0/0/0 extensive
Physical interface: xe-0/0/0, Enabled, Physical link is Up
<truncated>
  Traffic statistics:
   Input  bytes  :           68894515             19001536 bps
   Output bytes  :                128                    0 bps
   Input  packets:              87000                 3000 pps
   Output packets:                  2                    0 pps
<truncated>
```

The command shows that the T320 is receiving 3000 pps, as expected, from IXIA. Next, let's check the egress interface statistics. They should show 1000 pps each in the BE, AF, and EF queues. Here are the statistics for egress interface xe-4/1/0 on the T320:

```
[edit]
lab@Nuts-re0# run show interfaces xe-4/1/0 extensive
Physical interface: xe-4/1/0, Enabled, Physical link is Up
<truncated>
  Traffic statistics:
   Input  bytes  :                518                    0 bps
   Output bytes  :           87863257             19034152 bps
   Input  packets:                  5                    0 pps
   Output packets:             111009                 3001 pps
<truncated>
  Egress queues: 8 supported, 4 in use
  Queue counters:       Queued packets  Transmitted packets      Dropped packets
    0 BE                         74000                74000                    0
    1 EF                             0                    0                    0
    2 AF                         37000                37000                    0
    3 NC                             6                    6                    0
<truncated>
```

Of course, this traffic shows there is no EF traffic! All of the traffic is transmitted in either the BE or AF queue. It looks like the traffic stream that should be forwarded to the EF queue is actually going to the BE queue instead. Hmmm, this could impact sensitive applications.

Let's make sure the filter is properly applied to the logical interface index (IFL) of the xe-0/0/0 port. In order to do that on the T320, you need to log in to the FPC.

WARNING! Running commands on the linecards has the potential to cause impact, they should first be run in the lab and with JTAC instructions.

The following command can be used to log in to the FPC:

```
[edit]
lab@Nuts-re0# run start shell pfe network fpc0

GFPC platform (300Mhz MPC 755 processor, 256MB memory, 512KB flash)
```

Now, let's find which IFL index number maps to the xe-0/0/0.0 port. You can do this by reviewing the following output:

```
EGFPC0(Nuts-re0 vty)# show ifl brief
Index  Name                 Type          Encapsulation  Flags
-----  -------------------- ------------- -------------- ------
131073  .le1.0              Unspecified   Unspecified    0x0000000000000200
131074  .le2.0              Unspecified   Unspecified    0x0000000000000200
    0  .local..0            Unspecified   Unspecified    0x0000000000000010
    1  .local..1            Unspecified   Unspecified    0x0000000000000052
    2  .local..2            Unspecified   Unspecified    0x0000000000000052
   67  .local..3            Unspecified   Unspecified    0x0000000000000052
131075  .pfe.0              Unspecified   Unspecified    0x0000000000000040
   65  lo0.0                Unspecified   Unspecified    0x0000000000008052
   64  lo0.16384            Unspecified   Unspecified    0x0000000000008052
   66  lo0.16385            Unspecified   Unspecified    0x0000000000008052
   74  pc-0/0/0.16383       PIC Peer      PIC Peer       0x0000000000008010
   82  pc-2/0/0.16383       PIC Peer      PIC Peer       0x0000000000008010
   90  pc-4/0/0.16383       PIC Peer      PIC Peer       0x0000000000008010
   92  pc-4/1/0.16383       PIC Peer      PIC Peer       0x0000000000008010
   85  pc-7/0/0.16383       PIC Peer      PIC Peer       0x0000000000008010
   71  pfe-0/0/0.16383      Unspecified   Unspecified    0x0000000000008052
   69  pfe-1/0/0.16383      Unspecified   Unspecified    0x0000000000008052
   76  pfe-2/0/0.16383      Unspecified   Unspecified    0x0000000000008052
   89  pfe-4/0/0.16383      Unspecified   Unspecified    0x0000000000008052
   87  pfe-5/0/0.16383      Unspecified   Unspecified    0x0000000000008052
   78  pfe-6/0/0.16383      Unspecified   Unspecified    0x0000000000008052
   84  pfe-7/0/0.16383      Unspecified   Unspecified    0x0000000000008052
   70  pfh-0/0/0.16383      Unspecified   Unspecified    0x0000000000008052
   68  pfh-1/0/0.16383      Unspecified   Unspecified    0x0000000000008052
   75  pfh-2/0/0.16383      Unspecified   Unspecified    0x0000000000008052
   88  pfh-4/0/0.16383      Unspecified   Unspecified    0x0000000000008052
   86  pfh-5/0/0.16383      Unspecified   Unspecified    0x0000000000008052
   77  pfh-6/0/0.16383      Unspecified   Unspecified    0x0000000000008052
   83  pfh-7/0/0.16383      Unspecified   Unspecified    0x0000000000008052
   72  sp-1/1/0.16383       Services      Services       0x0000000000008010
   73  xe-0/0/0.0           Ethernet      Ethernet       0x000000000400c000
   91  xe-4/1/0.0           Ethernet      Ethernet       0x000000000400c000
```

You can see that interface xe-0/0/0.0 maps to IFL index number 73. So now let's use index number 73 in the next command to see which filters are applied to it:

```
EGFPC0(Nuts-re0 vty)# show ifl 73

Logical interface xe-0/0/0.0 (Index 73, Alias-Index 0 Peer-Index 0)
Channel Mode HYBRID  8
  Flags: (0x000000000400c000) Up SNMP-Traps Subset-iif
Addresses:
  Media address: Family: Unspecified (0), Chan: 0, Length: 0
IRB ifl BD index 65535
Reroute Ref: 0, Restore Ref: 0, LRID: 0
Residue Stats in:         0 out:         0
Protocols:
  Protocol: IPv4, MTU: 1500 bytes, Flags: 0x8000000200000000, Route table: 0
```

```
    Maximum labels: 0
    Input filter: 2, Output filter: 0, Interface class: 0, Dialer Filter: 0
    Input Simple Filter: 0, Output Simple Filter: 0
    Input implicit filters: None
    L2 Input policer: 0, L2 Output policer: 0
    Input policer: 0, Output policer: 0
    RPF fail-filter: 0, Reroute Ref: 0, Restore Ref: 0
    Service filters in: 0, 0, 0, 0, 0, 0, out: 0, 0, 0, 0, 0, 0, psf: 0
    Address(0): 192.168.1.5 (0x00) [primary] [192.168.1.4/30]
  Protocol: Multiservice, MTU: 65535 bytes, Flags: 0x0000000200000080, Route table: 0
    Maximum labels: 0
    Input filter: 0, Output filter: 0, Interface class: 0, Dialer Filter: 0
    Input Simple Filter: 0, Output Simple Filter: 0
    Input implicit filters: None
    L2 Input policer: 0, L2 Output policer: 0
    Input policer: 17000, Output policer: 0
    RPF fail-filter: 0, Reroute Ref: 0, Restore Ref: 0
Media:
  Type: Ethernet, Encapsulation: Ethernet (0x0000000E)
  MTU: 1514 bytes, Flags: 0x0000
Dependencies:
  Parent ifl index:     73
Storm control:
  BC: 0, UC: 0, Flags: 0x0
```

Input filter: 2 is highlighted in the output, showing that the xe-0/0/0.0 (IFL 73) interface has an input family inet firewall filter applied to it. This firewall filter has an index value of 2 associated with it. In order to view which firewall filter maps to filter index 2, you can use:

```
EGFPC0(Nuts-re0 vty)# show filter
Program Filters:
---------------
   Index     Dir     Cnt    Text     Bss  Name
--------  ------  ------  ------  ------  --------
       1     104       0      20      20  __default_bpdu_filter__
       2     208       0      88      88  MF-FILTER
   17000      52       0       4       4  __default_arp_policer__
   65280      52       0       4       4  __auto_policer_template__
   65281     104       0      16      16  __auto_policer_template_1__
   65282     156       0      32      32  __auto_policer_template_2__
   65283     208       0      48      48  __auto_policer_template_3__
   65284     260       0      64      64  __auto_policer_template_4__
   65285     312       0      80      80  __auto_policer_template_5__
   65286     364       0      96      96  __auto_policer_template_6__
   65287     416       0     112     112  __auto_policer_template_7__
   65288     468       0     128     128  __auto_policer_template_8__
```

So the filter index value of 2 is MF-FILTER, and that is the one that we configured. So, these FPC0 shell commands helped us verify that the correct firewall filter is assigned to xe-0/0/0.0. Let's exit the FPC0:

```
EGFPC0(Nuts-re0 vty)# exit
```

Okay, for the next troubleshooting step you can add counters to the MF classifier filter in order to see which terms the traffic streams are hitting.

Let's add the following counters to the filter configuration, one for each term:

```
[edit]
lab@Nuts-re0# set firewall family inet filter MF-
FILTER term Application_1 then count AF

[edit]
lab@Nuts-re0# set firewall family inet filter MF-
FILTER term Application_2 then count EF

[edit]
lab@Nuts-re0# set firewall family inet filter MF-
FILTER term DEFAULT then count DEFAULT

[edit]
lab@Nuts-re0# show firewall
family inet {
    filter MF-FILTER {
        term Application_1 {
            from {
                destination-address {
                    100.100.100.1/32;
                }
                protocol [ udp tcp ];
                destination-port 10000;
            }
            then {
                count AF;
                forwarding-class AF;
            }
        }
        term Application_2 {
            from {
                destination-address {
                    100.100.100.200/32;
                }
                protocol esp;
                port [ 50 51 500 ];
            }
            then {
                count EF;
                forwarding-class EF;
            }
        }
        term DEFAULT {
            then {
                count DEFAULT;
                accept;
            }
        }
    }
}

[edit]
lab@Nuts-re0# commit
commit complete
```

Okay that's commited. Next, let's check to see which counters are increasing, which would indicate that packets are hitting that term:

```
[edit]
lab@Nuts-re0# run show firewall filter MF-FILTER

Filter: MF-FILTER
Counters:
Name                                                Bytes              Packets
AF                                               24512912                31844
DEFAULT                                          49091854                63688
EF                                                      0                    0

[edit]
lab@Nuts-re0# run show firewall filter MF-FILTER

Filter: MF-FILTER
Counters:
Name                                                Bytes              Packets
AF                                               28003393                36344
DEFAULT                                          56035556                72688
EF                                                      0                    0

[edit]
lab@Nuts-re0# run show firewall filter MF-FILTER

Filter: MF-FILTER
Counters:
Name                                                Bytes              Packets
AF                                               31558923                40844
DEFAULT                                          62978514                81688
EF                                                      0                    0

[edit]
lab@Nuts-re0# run show firewall filter MF-FILTER

Filter: MF-FILTER
Counters:
Name                                                Bytes              Packets
AF                                               38488329                49844
DEFAULT                                          76926908                99688
EF                                                      0                    0
```

Notice that the EF counter is not counting in the firewall counter statistics above, which indicates that no packets are matching that term and therefore are not being assigned to the EF queue.

So now you need to find out why the packets are not matching the Application_2 term. Let's check the filter:

```
term Application_2 {
    from {
        destination-address {
            100.100.100.200/32;
        }
        protocol esp;
        port [ 50 51 500 ];
    }
    then {
        count EF;
        forwarding-class EF;
    }
}
```

And here you can see the problem is in the configuration of the `Application_2` term itself. Notice the matching conditions of destination-address, protocol esp, and port number criteria – the port criteria should be used with TCP or UDP, instead of ESP, because the ESP protocol does not have any ports. So no traffic could possibly match this term. The traffic is instead going to the default term and as a result using the BE queue.

The correct configuration for the `Application_2` term should be:

```
[edit]
lab@Nuts-re0# delete firewall family inet filter MF-
FILTER term Application_2 from port

[edit]
lab@Nuts-re0# show firewall family inet filter MF-FILTER
term Application_1 {
    from {
        destination-address {
            100.100.100.1/32;
        }
        protocol [ udp tcp ];
        destination-port 10000;
    }
    then {
        count AF;
        forwarding-class AF;
    }
}
term Application_2 {
    from {
        destination-address {
            100.100.100.200/32;
        }
```

```
            protocol esp;
        }
        then {
            count EF;
            forwarding-class EF;
        }
    }
    term DEFAULT {
        then {
            count DEFAULT;
            accept;
        }
    }

[edit]
lab@Nuts-re0# commit
commit complete
```

After the correct configuration is applied, you can see traffic hitting the Application_2 (EF) term using the existing firewall filter counters:

```
[edit]
lab@Nuts-re0# run show firewall filter MF-FILTER

Filter: MF-FILTER
Counters:
Name                                                Bytes              Packets
AF                                              132684373               171506
DEFAULT                                         261566759               338473
EF                                                3563790                 4539

[edit]
lab@Nuts-re0# run show firewall filter MF-FILTER

Filter: MF-FILTER
Counters:
Name                                                Bytes              Packets
AF                                              139692069               180506
DEFAULT                                         268540716               347473
EF                                               10522641                13539

[edit]
lab@Nuts-re0# run clear firewall all

[edit]
lab@Nuts-re0# run show firewall filter MF-FILTER

Filter: MF-FILTER
Counters:
Name                                                Bytes              Packets
AF                                                1274865                 1690
DEFAULT                                           1327938                 1690
EF                                                1314269                 1690
```

Let’s also verify that packets are being sent out of the egress xe-4/1/0 interface in the intended queues:

```
[edit]
lab@Nuts-re0# run show interfaces xe-4/1/0 extensive
Physical interface: xe-4/1/0, Enabled, Physical link is Up
<truncated>
  Traffic statistics:
   Input  bytes  :              5748                    0 bps
   Output bytes  :        1233531059             18702128 bps
   Input  packets:                59                    0 pps
   Output packets:           1559997                 3000 pps
<truncated>
  Egress queues: 8 supported, 4 in use
  Queue counters:       Queued packets  Transmitted packets      Dropped packets
    0 BE                    938827               938827                    0
    1 EF                    101132               101132                    0
    2 AF                    519979               519979                    0
    3 NC                        59                   59                    0
<truncated>
```

Summary

In this recipe we were able to identify that packets were not using the appropriate term in a MF Classifier. This could have a large impact in some production environments.

First, we used the `show interfaces extensive <interface-name>` command to check the queue statistics on the egress interface, and observed that no packets were being sent out of the EF queue – this was unexpected.

In Junos, MF Classifiers are applied through a firewall filter on the interface, so we verified that the filter was applied on the ingress interface. This ensured that the filter was at least being evaluated when ingress packets arrived.

The firewall filter *was* being evaluated but the ESP packets were still not being sent out of the EF queue. So as a next troubleshooting step we applied a counter to each of the MF Classifier firewall filter terms. This showed us that the Application_2 term did not match any of the packets. We took a closer look at the configuration of the Application_2 filter term, and identified a configuration mistake. Once this was corrected, the packets were going out of the correct queue.

Using firewall filter counters can be very useful in many different scenarios. It can help determine if a firewall filter term is working as intended or not, and in our case, it showed us that the problem was related to a specific filter term.

Recipe 6

Class of Service Rewrite Rule Not Correctly Marking

This receipe shows the troubleshooting steps necessary to identify a problem that occurs when the rewrite rule is not correctly marking the Type of Service (ToS) bits of traffic on a Juniper router. The steps set forth in this recipe will help you to identity the problem and the ways it can be fixed.

Problem

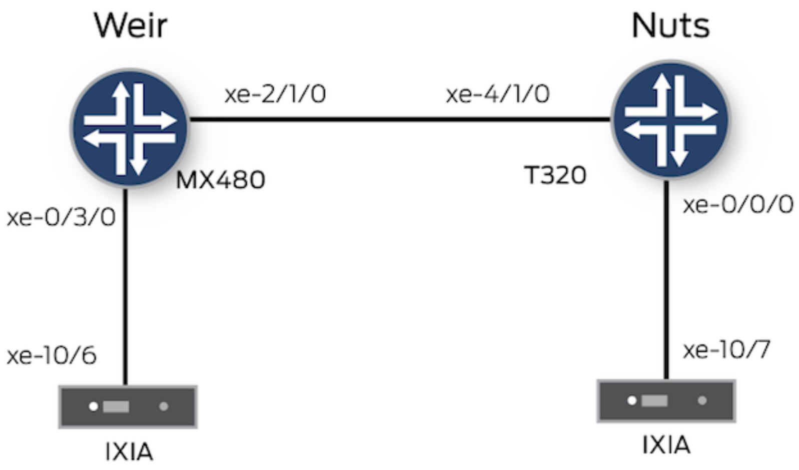
The class of service rewrite rule is not rewriting the ToS bits of the packets on egress. As a result, the packets are not classified correctly on the next hop router.

There are several reasons why this problem might happen, including a configuration problem, a software problem, or even a hardware problem, and they will be reviewed in this recipe.

Solution

When network engineers observe that packets are not being rewritten on a router's egress interface with a defined DSCP marking, it can cause packets to be handled improperly on other devices in the network. It is important that the user-defined rewrite rules are operating as intended as they will have an impact on certain high priority traffic. Let's take a look at this scenario.

Our trusty lab routers, Weir and Nuts, are MX480 and T320, respectively. They are both running Junos 11.4R1.14 and they are each connected to an IXIA Traffic Generator.



In this recipe's topology, the IXIA Traffic Generator is sending a total of about 150,000 frames per second from port xe-10/6, through Weir and Nuts, to IXIA port xe-10/7. Bandwidth would not be a problem here, since all interfaces are 10Gbps and the 150,000 frames per second would not exceed 10Gbps. So there shouldn't be any packet drops due to bandwidth being exceeded on the link. Three streams of traffic are being sent:

- One stream is 50,000 frames per second and has a default DSCP bit marking of best-effort (BE), or "000000".
- The second stream is 50,000 frames per second and has a default DSCP bit marking of assured-forwarding (AF11), or "001010".
- The final stream is also being sent at a rate of 50,000 frames per second and has a default DSCP bit marking of expedited-forwarding (EF), or "101110".

A user-defined DSCP rewrite rule is applied to the Weir router's xe-2/1/0 interface, which is the egress interface facing Nuts. This rewrite rule is intended to re-mark the assured-forwarding (AF11) "001010" value from packets to the same value that expedited-forwarding (EF) is using, "101110". So the intent is that the next hop router, Nuts, will classify the previously-marked AF11 traffic and map it to the EF queue to go along with the original EF marked traffic. The rewrite rule configuration is:

```
class-of-service {
   <…>
   interfaces {
      <…>
      xe-2/1/0 {
         scheduler-map CORE;
         unit 0 {
            classifiers {
               dscp CLASSIFIER_1;
            }
            rewrite-rules {
               dscp REWRITE_RULE_1;
            }
         }
      }
   }
   rewrite-rules {
      dscp REWRITE_RULE_1 {
         forwarding-class BE {
            loss-priority low code-point 000000;
         }
         forwarding-class AF {
            loss-priority low code-point 101110;
         }
         forwarding-class EF {
            loss-priority high code-point 101110;
         }
         forwarding-class NC {
            loss-priority high code-point 110000;
         }
      }
   }
```

Notice in the configuration above that the custom DSCP rewrite rule named `REWRITE_RULE_1` is being referenced under the xe-2/1/0.0 logical interface. It is then defined under the [edit class-of-service rewrite-rules] hierarchy. Also notice that the forwarding-class AF code-points are the same when compared to the EF forwarding-class, and this should take the AF11 marked traffic that is being sent out of the assured-forwarding queue and rewrite the DSCP bits to the same ones that are are being used for the EF queue.

The sequence of events that takes place here are: first, that the Weir router receives the AF11 traffic from IXIA; second, that Weir classifies the AF11 traffic via the ingress interface's BA Classifier and maps it to the assured forwarding queue at the egress interface xe-2/1/0.0; and third, that the egress interface xe-2/1/0.0 also has this rewrite rule applied to it and rewrites the code-points of the AF11 traffic to "101110", which happens to be the same code-points that the EF traffic uses. Such traffic on Weir is still being forwarded out of the assured forwarding queue, so the next hop router Nuts should then see such traffic with the new DSCP bit marking "101110" instead of AF11.

Let's check the egress xe-2/1/0 interface on Weir to see if it is correctly showing transmitted packets in the BE, EF, and AF queues:

```
{master}[edit]
lab@Weir-re0# run show interfaces xe-2/1/0 extensive
Physical interface: xe-2/1/0, Enabled, Physical link is Up
<...>
  Egress queues: 8 supported, 4 in use
  Queue counters:       Queued packets  Transmitted packets      Dropped packets
    0 BE                       1080011              1080011                    0
    1 EF                       1080011              1080011                    0
    2 AF                       1080011              1080011                    0
    3 NC                             3                    3                    0
  <...>
```

Let's look at the next router, Nuts. We should see only BE and EF traffic on Nuts since the Weir router is remarking the AF11 traffic to use the same code-points as EF uses. Here are the ingress interface statistics on Nuts:

```
{master}[edit]
lab@Nuts-re0# run show interfaces xe-4/1/0 extensive
Physical interface: xe-4/1/0, Enabled, Physical link is Up
<…>
  Ingress queues: 8 supported, 4 in use
  Queue counters:       Queued packets  Transmitted packets      Dropped packets
    0 BE                       1750019              1750019                    0
    1 EF                       1750019              1750019                    0
    2 AF                       1750019              1750019                    0
    3 NC                             4                    4                    0
```

The xe-4/1/0 ingress traffic in the BE, EF, and AF queues shows that the rewrite rule applied on Weir is not working correctly because packets from Weir are still classified in the AF queue in Nuts! You should see only packets on Nuts in the BE and the EF queues. As a result, the mere existence of packets for all three classifications shows that there is an issue with the rewrite rule on Weir.

Let's see if the rewrite rule is applied correctly to the xe-2/1/0 egress interface on Weir. The `show class-of-service interface <interface-name>` command is useful because it shows any scheduler-map, rewrite-rule, and classifier applied to the interface:

```
{master}[edit]
lab@Weir-re0# run show class-of-service interface xe-2/1/0
Physical interface: xe-2/1/0, Index: 145
Queues supported: 8, Queues in use: 4
  Scheduler map: CORE, Index: 46629
  Congestion-notification: Disabled
  Logical interface: xe-2/1/0.0, Index: 196612
    Object                  Name                   Type                    Index
    Rewrite                 REWRITE_RULE_1         dscp                    57803
    Classifier              CLASSIFIER_1           dscp                    42245
```

Notice in the output here that `REWRITE_RULE_1` is applied to the xe-2/1/0.0 interface as expected. Next, the show class-of-service rewrite-rule name <rewrite-rule-name> command is used to see the details of how the rewrite rule is configured:

```
{master}[edit]
lab@Weir-re0# run show class-of-service rewrite-rule name REWRITE_RULE_1
Rewrite rule: REWRITE_RULE_1, Code point type: dscp, Index: 57803
  Forwarding class                    Loss priority       Code point
  BE                                  low                 000000
  EF                                  high                101110
  AF                                  low                 101110
  NC                                  high                110000
```

There doesn't appear to be any problem here. Notice that the AF forwarding-class is configured to rewrite the appropriate packets to use code-point "101110", which is the same as the EF forwarding class. This matches what we have configured.

To confirm that the packets are arriving on the Nuts xe-4/1/0 ingress interface without the modified DSCP value, you can use firewall filter counters to count packets with specific DSCP markings.

The following firewall filter was configured and applied on the xe-4/1/0 Nuts as an ingress filter:

```
{master}[edit]
lab@Nuts-re0# show firewall family inet filter TEST_DSCP
interface-specific;
term 1 {
    from {
        dscp be;
    }
    then {
        count dscp-be;
        accept;
    }
}
term 2 {
    from {
        dscp af11;
    }
    then {
        count dscp-af11;
        accept;
    }
}
```

```
term 3 {
    from {
        dscp ef;
    }
    then {
        count dscp-ef;
        accept;
    }
}
term default {
    then accept;
}

{master}[edit]
lab@Nuts-re0# show interfaces xe-4/1/0
unit 0 {
    family inet {
        filter {
            input TEST_DSCP;
        }
        address 10.1.1.2/30;
    }
}
```

While checking the firewall filter counters, you can see that they are incrementing for BE, AF11, and EF, confirming that the AF11 packets are not being rewritten on Weir's egress xe-2/1/0 interface before arriving on the Nuts xe-4/1/0 ingress interface:

```
{master}[edit]
lab@Nuts-re0# run show firewall

Filter: __default_bpdu_filter__

Filter: TEST_DSCP-xe-4/1/0.0-i
Counters:
Name                                                Bytes              Packets
dscp-af11-xe-4/1/0.0-i                         2312677308              2992363
dscp-be-xe-4/1/0.0-i                           2313928902              2992362
dscp-ef-xe-4/1/0.0-i                           2313431747              2992363

{master}[edit]
lab@Nuts-re0# run show firewall

Filter: __default_bpdu_filter__

Filter: TEST_DSCP-xe-4/1/0.0-i
Counters:
Name                                                Bytes              Packets
dscp-af11-xe-4/1/0.0-i                         2486643340              3217364
dscp-be-xe-4/1/0.0-i                           2487867577              3217364
dscp-ef-xe-4/1/0.0-i                           2487193503              3217364
```

If the rewrite rule on Weir is working correctly, then we shouldn't see any AF11 packets incrementing in the above firewall filter counter. The firewall filter counter can be used for investigating a range of different issues and is a very valuable troubleshooting tool. In our case, with this recipe, it helped verify that there is a problem with the rewrite rule on the Weir router that is applied on the xe-2/1/0 interface.

Now let's check the programming at the PFE level to make sure the rewrite rule is programmed correctly on the xe-2/1/0 interface:

```
{master}[edit]
lab@Weir-re0# run show class-of-service forwarding-table rewrite-rule mapping
Interface       Index    Table index  Type
xe-2/1/0.0       196612    57803      DSCP
```

And you can see that the logical interface xe-2/1/0.0 has an index value of 196612. The rewrite-rule index is 57803.

Another way to verify the index number of logical interface xe-2/1/0.0 is shown below. You can see that it is mapped to IFL index 196612.

WARNING! Running commands on the linecards has the potential to cause impact: the commands should first be run in the lab, or with JTAC instructions.

```
{master}[edit]
lab@Weir-re0# run start shell pfe network fpc2

ADPC platform (1200Mhz MPC 8548 processor, 1024MB memory, 512KB flash)

ADPC2(Weir-re0 vty)# show ifl brief
Index  Name                  Type          Encapsulation   Flags
-----  -------------------- ------------- --------------- ------
<...>
196612  xe-2/1/0.0            Ethernet       Ethernet        0x000000000400c000
```

Next, check the rewrite rule and see that it is, in fact, programmed correctly, to the right interface:

```
ADPC2(Weir-re0 vty)# show cos rw-table
    id      num_entry              type       ifl assigned
    --      ---------              ----       ------------
 57803          4                  DSCP        196612
```

And the rewrite rule 57803 is assigned to the interface ifl 196612 (xe-2/1/0.0), which is correct. The following command shows the rewrite rule is applied to interface index 196612:

```
ADPC2(Weir-re0 vty)# show cos rw-table ifl 196612
cos_ifl: 196612
nh_rw_type                 : IP or DSCP rw table (0)
  nh_cos_table_id          : 57803
  nh_cos_multi_fc          : TRUE
  nh_cos_dscp_rw           : TRUE
  nh_cos_inet_prec_rw      : FALSE
  nh_cos_tag_rw            : FALSE
  nh_cos_ieee_rw           : FALSE
  nh_cos_bind_flag         : 0x00000000
  nh_cos_table[  0]        : 0x80
  nh_cos_table[  2]        : 0x8a
  nh_cos_table[193]        : 0xae
  nh_cos_table[195]        : 0xb0
  rw_table_id              : 57803
  rw_table_num_entries     : 4
  rw_table_type            : DSCP (1)
        cos_ifl info:
```

Next, let's check the programming for this particular rewrite rule to see if it is correct:

```
ADPC2(Weir-re0 vty)# show cos rw-table 57803

    FC  DP   CP      FC  DP   CP      FC  DP   CP      FC  DP   CP
    ------------     ------------     ------------     ------------
     0   0  0x00      1   3  0x2e      2   0  0x0a      3   3  0x30
```

This output shows forwarding-class (FC) 0 is best-effort, and has a rewrite code-point (CP) value of "0x00", which is the value "000000" that is configured. The FC 1 is expedited-forwarding, and has a code-point value of "0x2e", which is the "101110" that is configured for EF. Now, the problem is that FC 2, assured-forwarding, has a code-point value of "0x0a", which is not the same as the one for EF.

So here is where the issue lies. The configured re-write rule is configured and applied in the PFE, but the assured-forwarding queue is not rewriting the packets' DSCP value to "0x2e" or "101110" as it should be. Instead, it is leaving the DSCP value field as "0x0a".

There are many actions that can be attempted to fix this problem. For example, you can try restarting the class-of-service daemon (cosd), restarting the FPC, bouncing the interface, removing and re-adding the rewrite rule via configuration, performing an RE switchover, or rebooting.

To fix it, let's remove the rewrite rule, commit the configuration, then re-add the rewrite rule followed by another commit. This forces the rewrite rule to be removed from the PFE and re-added back to the PFE again. Follow along:

```
{master}[edit]
lab@Weir-re0# show class-of-service interfaces
xe-0/3/0 {
    scheduler-map CORE;
    unit 0 {
        classifiers {
            dscp CLASSIFIER_1;
        }
    }
}
xe-2/1/0 {
    scheduler-map CORE;
    unit 0 {
        classifiers {
            dscp CLASSIFIER_1;
        }
        rewrite-rules {
            dscp REWRITE_RULE_1;
        }
    }
}

{master}[edit]
lab@Weir-re0# deactivate class-of-service interfaces xe-2/1/0 unit 0 rewrite-rules

{master}[edit]
lab@Weir-re0# commit
re0:
configuration check succeeds
re1:
commit complete
re0:
commit complete

{master}[edit]
lab@Weir-re0# show class-of-service interfaces
xe-0/3/0 {
    scheduler-map CORE;
    unit 0 {
        classifiers {
            dscp CLASSIFIER_1;
        }
    }
}
xe-2/1/0 {
    scheduler-map CORE;
    unit 0 {
        classifiers {
            dscp CLASSIFIER_1;
        }
        inactive: rewrite-rules {
            dscp REWRITE_RULE_1;
        }
    }
}

{master}[edit]
lab@Weir-re0# activate class-of-service interfaces xe-2/1/0 unit 0 rewrite-rules

{master}[edit]
```

```
lab@Weir-re0# commit
re0:
configuration check succeeds
re1:
commit complete
re0:
commit complete

{master}[edit]
lab@Weir-re0# show class-of-service interfaces
xe-0/3/0 {
    scheduler-map CORE;
    unit 0 {
        classifiers {
            dscp CLASSIFIER_1;
        }
    }
}
xe-2/1/0 {
    scheduler-map CORE;
    unit 0 {
        classifiers {
            dscp CLASSIFIER_1;
        }
        rewrite-rules {
            dscp REWRITE_RULE_1;
        }
    }
}
```

To make sure this worked, let's clear the firewall filter statistics on the Nuts router and check it again:

```
{master}[edit]
lab@Nuts-re0# run show firewall

Filter: __default_bpdu_filter__

Filter: TEST_DSCP-xe-4/1/0.0-i
Counters:
Name                                                    Bytes              Packets
dscp-af11-xe-4/1/0.0-i                                      0                    0
dscp-be-xe-4/1/0.0-i                                 32240040                41851
dscp-ef-xe-4/1/0.0-i                                 64716791                83702
```

Now packets are no longer coming in with the AF11 DSCP marking, as expected. You can see they are now being correctly rewritten to the EF DSCP value.

The same PFE shell command on Weir shows that the FC 2 correctly has the same DSCP code-point rewrite value, as the FC 1.

```
ADPC2(Weir-re0 vty)# sh cos rw-table 57803

    FC  DP   CP      FC  DP   CP      FC  DP   CP      FC  DP   CP
    ------------     ------------     ------------     ------------
     0   0  0x00      1   3  0x2e      2   0  0x2e      3   3  0x30
```

Summary

The class of service rewrite rule was not working correctly. As a result, the next hop router was not classifying the AF11 packets as intended. In a real world scenario, network operations engineers may notice that this could cause unexpected packet drops or unexpected traffic patterns shown in graphs for particular forwarding classes. It is always important to quickly identify the cause of the problem to prevent a possible impact to traffic.

First, the problem was noticed on the next hop router Nuts by checking the `show interfaces extensive` command, which showed that this router was receiving ingress traffic from Weir that was associated with the BE, AF, and EF forwarding classes. This was unexpected because Nuts should only receive traffic associated with the BE or EF forwarding classes, since Weir was supposed to rewrite the AF traffic to the same as the EF.

The configuration of the rewrite rule on Weir was then checked. The `show class-of-service interface <interface-name>` command is valuable because it shows us what classifiers, rewrite-rules, and scheduler-maps are configured for a particular interface.

Next, a firewall filter was added on Nuts to count traffic matching the DSCP bit values that we are using. We observed that Nuts was receiving traffic marked with BE, AF11, and EF bits. This was unexpected because Weir should have been rewriting the AF11 bits to the same as the EF. Using this firewall filter counter helped us verify that the problem was with the Weir rewrite rule.

So we focused on the rewrite rule on the Weir egress interface and logged into the PFE and confirmed that the correct rewrite rule was applied to the egress interface. The programming was incorrect, however. The rewrite rule programming on the PFE does not match the configuration that we applied.

To fix the problem, we removed the rewrite rule from the Weir egress interface then applied it back again. After this, the firewall filter on Nuts no longer showed any AF11 packets and the PFE on Weir reflected the correct programming.

Recipe 7

Class of Service Scheduler Map Not Working

The recipe used here demonstrates some of the troubleshooting steps that can be used to look at scheduler map problems, and the CLI commands used in this recipe can be used for a variety of other issues as well.

Problem

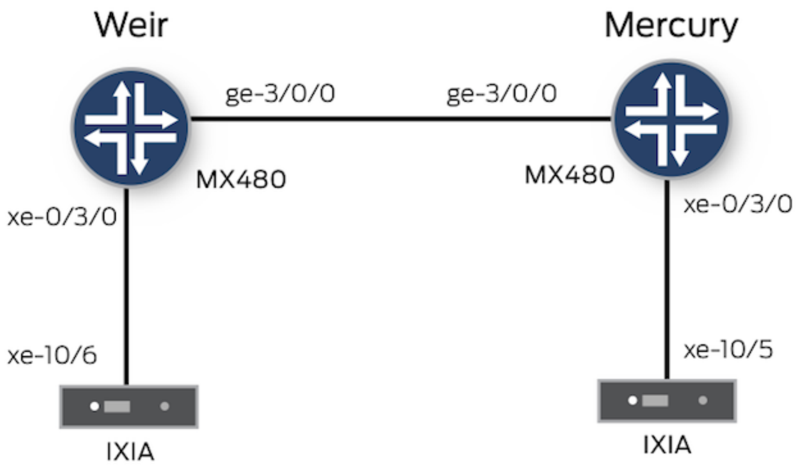
The class of service scheduler map is not working correctly. There could be many symptoms and causes associated with scheduler map problems in class of service: it could happen due to a configuration mistake, software problem, hardware problem, or compatibility problem. The symptoms of class of service scheduler map problems might be dropped packets, traffic delay, jitter, poor voice call quality, dropped voice calls, and choppy or stalled video. This recipe focuses on a problem related to a class of service scheduler map causing packet drops.

Note that a problem with the class of service scheduler maps can happen in any environment. Scheduler maps are commonly used to prioritize certain traffic, for example, voice traffic is much more delay sensitive as compared to certain data traffic. Customers may have specific applications that they want to prioritize over other traffic, so it is important to have a correct class of service configuration to accommodate. Let's begin!

Solution

Packet drops are noticed in the high-priority Expedited Forwarding queue but not in the low-priority Best Effort queue. We expect that the router should prioritize the traffic and drop the packets in the low priority queue first – this could have a big impact on certain customer applications in a real world scenario!

Our lab routers, Weir and Nuts, are MX480 and T320, respectively. Both are running Junos 11.4R1.14 and connected to an IXIA Traffic Generator.



The IXIA Traffic Generator is sending three traffic streams from port xe-10/6 to xe-10/5:

Stream #1

- Source IP - 192.168.1.2
- Destination IP range – 101.101.101.1 through 101.101.101.100
- Protocol – UDP
- Frame Size – Random, from 64 to 1500
- TOS – DSCP 000000
- Rate – 765Mbps

Stream #2

- Source IP - 192.168.1.2
- Destination IP range – 101.101.101.1 through 101.101.101.100
- Protocol – UDP
- Frame Size – Random, from 64 to 1500
- TOS – DSCP 001010
- Rate – 180Mbps

Stream #3

- Source IP - 192.168.1.2
- Destination IP range – 101.101.101.1 through 101.101.101.100
- Protocol – UDP
- Frame Size – Random, from 64 to 1500
- TOS – DSCP 101110
- Rate – 135Mbps

Notice the DSCP bit markings are different for each of the three traffic streams. This allows the routers to classify these packets and forward them out three different queues. For example, Stream #1 uses the Best Effort (BE) queue, Stream #2 uses the Assured Forwarding (AF) queue, and Stream #3 uses the Expedited Forwarding (EF) queue.

This can be confirmed by looking at the code-point aliases and the classifier configuration on the router. Let's show the code-point aliases:

```
master}[edit]
lab@Weir-re0# run show class-of-service code-point-aliases dscp
Code point type: dscp
  Alias                 Bit pattern
  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
  cs6                   110000
  cs7                   111000
  ef                    101110
  nc1                   110000
  nc2                   111000
```

Notice that the bit pattern “000000” is mapped to the alias “be”. The bit pattern “001010” is mapped to alias “af11” and bits “101110” is mapped to “ef”.

Next, let’s check the ingress interface on the Weir router to see which DSCP classifier it is using:

```
{master}[edit]
lab@Weir-re0# run show class-of-service interface xe-0/3/0
Physical interface: xe-0/3/0, Index: 243
Queues supported: 8, Queues in use: 4
  Scheduler map: CORE, Index: 46629
  Congestion-notification: Disabled

  Logical interface: xe-0/3/0.0, Index: 373
    Object                  Name                   Type                    Index
    Classifier              CLASSIFIER_1           dscp                    42245
```

In the CLI output below, the user-defined `CLASSIFIER_1` DSCP classifier is assigned to this interface.

Let’s check the CLASSIFIER_1 DSCP classifier configuration:

```
{master}[edit]
lab@Weir-re0# show class-of-service classifiers
      dscp CLASSIFIER_1 {
          forwarding-class BE {
              loss-priority low code-points 000000;
          }
          forwarding-class AF {
              loss-priority low code-points 001010;
          }
          forwarding-class EF {
              loss-priority high code-points 101110;
          }
          forwarding-class NC {
              loss-priority high code-points 110000;
          }
      }
```

And this output confirms that the bit markings map to the correct forwarding-class.

Now that the traffic will be placed in three different queues, let's look at the scheduler map configuration on the Weir router to see how the traffic will be prioritized. First, let's check to see which scheduler map is configured on the egress ge-3/0/0 interface:

```
{master}[edit]
lab@Weir-re0# show class-of-service interfaces
        xe-0/3/0 {
            scheduler-map CORE;
            unit 0 {
                classifiers {
                    dscp CLASSIFIER_1;
                }
            }
        }
        ge-3/0/0 {
            scheduler-map CORE;
            unit 0 {
                classifiers {
                    dscp CLASSIFIER_1;
                }
            }
        }
```

As shown here, the `CORE` scheduler-map is applied to the ge-3/0/0 egress interface. Next, let's look at the scheduler-map and associated schedulers configuration:

```
{master}[edit]
lab@Weir-re0# show class-of-service
    <truncated>
    scheduler-maps {
        CORE {
            forwarding-class BE scheduler CORE-BE;
            forwarding-class AF scheduler CORE-AF;
            forwarding-class EF scheduler CORE-EF;
            forwarding-class NC scheduler CORE-NC;
        }
    }
    schedulers {
        CORE-BE {
            transmit-rate 765m;
            buffer-size {
                remainder;
            }
            priority medium-low;
        }
        CORE-AF {
            transmit-rate percent 18;
            buffer-size percent 18;
            priority medium-high;
        }
        CORE-EF {
            transmit-rate 135m;
            buffer-size percent 10;
            priority high;
        }
        CORE-NC {
            transmit-rate percent 5;
            buffer-size percent 5;
            priority strict-high;
        }
    }
}
```

The BE queue has medium-low priority, the AF queue has medium-high priority, and the EF queue has high priority.

The BE queue is assigned 765Mbps and it's sending 765Mbps of this traffic. Since the other queues are higher priority, one would expect to see some drops in the BE queue while more of the AF and EF packets are forwarded.

The interface statistics can be checked with the `show interfaces <interface> extensive` command. Notice that the output packet rate is close to line rate, and that there are packet drops in the high priority EF queue, but not in the lower priority BE queue!

```
{master}
lab@Weir-re0> show interfaces ge-3/0/0 extensive
Physical interface: ge-3/0/0, Enabled, Physical link is Up
  Interface index: 147, SNMP ifIndex: 1546, Generation: 395
  Link-level type: Ethernet, MTU: 1514, Speed: 1000mbps, BPDU Error: None, MAC-
REWRITE Error: None,
  Loopback: Disabled, Source filtering: Disabled, Flow control: Enabled, Auto-
negotiation: Enabled,
  Remote fault: Online
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x4000
  Link flags     : None
  CoS queues     : 8 supported, 4 maximum usable queues
  Schedulers     : 0
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: 78:19:f7:50:23:de, Hardware address: 78:19:f7:50:23:de
  Last flapped   : 2012-03-04 10:36:34 PST (01:43:35 ago)
  Statistics last cleared: 2012-03-04 12:19:43 PST (00:00:26 ago)
  Traffic statistics:
   Input  bytes  :                  304                    0 bps
   Output bytes  :           3514512566           1069633992 bps
   Input  packets:                    4                    0 pps
   Output packets:              4546691               172663 pps
   IPv6 transit statistics:
    Input  bytes  :                0
    Output bytes  :                0
    Input  packets:                0
    Output packets:                0
  Dropped traffic statistics due to STP State:
   Input  bytes  :                 0
   Output bytes  :                 0
   Input  packets:                 0
   Output packets:                 0
  Input errors:
    Errors: 0, Drops: 0, Framing errors: 0, Runts: 0, Policed discards: 0, L3
incompletes: 0, L2 channel errors: 0,
    L2 mismatch timeouts: 0, FIFO errors: 0, Resource errors: 0
  Output errors:
    Carrier transitions: 0, Errors: 0, Drops: 0, Collisions: 0, Aged packets: 0, FIFO
errors: 0,
    HS link CRC errors: 0, MTU errors: 0, Resource errors: 0
  Egress queues: 8 supported, 4 in use
  Queue counters:       Queued packets  Transmitted packets      Dropped packets
    0 BE                       3022309              3022310                    0
    1 AF                        711132               408321               302809
    2 EF                        533352               412124               121226
    3 NC                             3                    3                    0
```

Another way to check the drop counters is to look at the `show interfaces queue <interface>` command as shown below. This command is very useful for troubleshooting because it displays detailed per-queue statistics on an interface. Notice the RED drops in the AF and EF queues instead of the BE queue:

```
{master}
lab@Weir-re0> show interfaces queue ge-3/0/0
Physical interface: ge-3/0/0, Enabled, Physical link is Up
  Interface index: 147, SNMP ifIndex: 1546
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Queue: 0, Forwarding classes: BE
  Queued:
    Packets              :             16320465                120892 pps
    Bytes                :          12909086950             763605216 bps
  Transmitted:
    Packets              :             16320464                 120892 pps
    Bytes                :          12909086438             763605216 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                    0                     0 pps
     Low                 :                    0                     0 pps
     Medium-low          :                    0                     0 pps
     Medium-high         :                    0                     0 pps
     High                :                    0                     0 pps
    RED-dropped bytes    :                    0                     0 bps
     Low                 :                    0                     0 bps
     Medium-low          :                    0                     0 bps
     Medium-high         :                    0                     0 bps
     High               :                 0                 0 bps
Queue: 1, Forwarding classes: AF
  Queued:
    Packets             :           3840109             28446 pps
    Bytes               :        3037763192         179862336 bps
  Transmitted:
    Packets             :           2205956              16433 pps
    Bytes               :        1776755201         106008912 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :           1634152             12013 pps
     Low                :           1634152             12013 pps
     Medium-low         :                 0                 0 pps
     Medium-high        :                 0                 0 pps
     High               :                 0                 0 pps
    RED-dropped bytes   :        1261007479          73853424 bps
     Low                :        1261007479          73853424 bps
     Medium-low         :                 0                 0 bps
     Medium-high        :                 0                 0 bps
     High               :                 0                 0 bps
Queue: 2, Forwarding classes: EF
  Queued:
    Packets             :           2880085             21336 pps
    Bytes               :        2278428718         135085304 bps
  Transmitted:
    Packets             :           2225704              16570 pps
    Bytes               :        1774220031         105772240 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :            654379              4766 pps
     Low                :                 0                 0 pps
     Medium-low         :                 0                 0 pps
     Medium-high        :                 0                 0 pps
     High               :            654379              4766 pps
    RED-dropped bytes   :         504207663          29313064 bps
     Low                :                 0                 0 bps
     Medium-low         :                 0                 0 bps
     Medium-high        :                 0                 0 bps
     High               :         504207663          29313064 bps
Queue: 3, Forwarding classes: NC
  Queued:
    Packets             :                16                 0 pps
    Bytes               :              1572                 0 bps
  Transmitted:
    Packets             :                16                  0 pps
    Bytes               :              1572                 0 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                 0                 0 pps
     Low                :                 0                 0 pps
     Medium-low         :                 0                 0 pps
     Medium-high        :                 0                 0 pps
     High               :                 0                 0 pps
    RED-dropped bytes   :                 0                 0 bps
     Low                :                 0                 0 bps
     Medium-low         :                 0                 0 bps
     Medium-high        :                 0                 0 bps
     High               :                 0                 0 bps
```

To make sure that the scheduler-map `CORE` is applied to the interface as one would expect it to be, the `show class-of-service interface ge-3/0/0` command is used. Notice that it is applied to ge-3/0/0 as expected:

```
{master}[edit]
lab@Weir-re0# run show class-of-service interface ge-3/0/0
Physical interface: ge-3/0/0, Index: 147
Queues supported: 4, Queues in use: 4
  Scheduler map: CORE, Index: 46629
  Chassis scheduler map: <default-chassis>, Index: 4
  Congestion-notification: Disabled

  Logical interface: ge-3/0/0.0, Index: 65541
    Object                  Name                  Type                    Index
    Classifier              CLASSIFIER_1          dscp                    42245
```

To view more detailed information about the applied `CORE` scheduler-map, the `show class-of-service scheduler-map <name>` command can be used. This shows the individual schedulers that are applied to each forwarding-class within the scheduler-map:

```
{master}[edit]
lab@Weir-re0# run show class-of-service scheduler-map CORE
Scheduler map: CORE, Index: 46629

  Scheduler: CORE-BE, Forwarding class: BE, Index: 26391
    Transmit rate: 765000000 bps, Rate Limit: none, Buffer size: remainder, Buffer
Limit: none, Priority: medium-low
    Excess Priority: unspecified
    Drop profiles:
      Loss priority   Protocol    Index    Name
       Low             any             1    <default-drop-profile>
       Medium low      any             1    <default-drop-profile>
       Medium high     any             1    <default-drop-profile>
       High            any             1    <default-drop-profile>

  Scheduler: CORE-AF, Forwarding class: AF, Index: 26484
    Transmit rate: 18 percent, Rate Limit: none, Buffer size: 18 percent, Buffer Limit:
none, Priority: medium-high
    Excess Priority: unspecified
    Drop profiles:
      Loss priority   Protocol    Index    Name
       Low             any             1    <default-drop-profile>
       Medium low      any             1    <default-drop-profile>
       Medium high     any             1    <default-drop-profile>
       High            any             1    <default-drop-profile>

  Scheduler: CORE-EF, Forwarding class: EF, Index: 26612
    Transmit rate: 135000000 bps, Rate Limit: none, Buffer size: 10 percent, Buffer
Limit: none, Priority: high
    Excess Priority: unspecified
     Drop profiles:
        Loss priority   Protocol    Index    Name
        Low             any             1    <default-drop-profile>
        Medium low      any             1    <default-drop-profile>
        Medium high     any             1    <default-drop-profile>
        High            any             1    <default-drop-profile>

  Scheduler: CORE-NC, Forwarding class: NC, Index: 26257
    Transmit rate: 5 percent, Rate Limit: none, Buffer size: 5 percent, Buffer Limit:
none, Priority: strict-high
    Excess Priority: unspecified
     Drop profiles:
        Loss priority   Protocol    Index    Name
        Low             any             1    <default-drop-profile>
        Medium low      any             1    <default-drop-profile>
        Medium high     any             1    <default-drop-profile>
        High            any             1    <default-drop-profile>
```

So far the problem may not be obvious. After all, the scheduler-map "CORE" shows that it is applied to the interface in question, as expected. But let's go into the shell and make sure that the scheduler-map is *correctly* programmed in the PFE:

```
{master}[edit]
lab@Weir-re0# run start shell pfe network fpc3
```

WARNING! This book is about using all the tools at your disposal. You know that using shell commands is not supported or endorsed, but let's proceed like careful, prudent engineers who want to solve a problem.

First, let's get the physical interface (IFD) index number for the ge-3/0/0 link in question, since the scheduler map is applied to the physical interface instead of the logical interface:

```
ADPC platform (1200Mhz MPC 8548 processor, 1024MB memory, 512KB flash)

ADPC3(Weir-re1 vty)# show ifd brief
Index  Name                 Type         Flags  Slot   State
-----  -------------------- -----------  ------ -----  ------
<truncated>
  147  ge-3/0/0             Ethernet     0x0000000000008000     3  Up
 <truncated>
```

From the PFE output, you can see that the physical interface (IFD) ge-3/0/0 has an index number of 147 assigned to it.

Next, let's check the class-of-service scheduling-policy that is assigned to this IFD index number 147, which maps to ge-3/0/0, the interface in question:

```
ADPC3(Weir-re1 vty)# show f-cos scheduling-policy 147
Output scheduling policy
========================

ifd ge-3/0/0 (147)
   ifd   Q  tx-BW delay-BW  sched   rate tcp/plp tcp/plp tcp/plp tcp/plp excess  excess  
shaping
  index num  (%)    (%)    priority cntl  (0/0)   (0/1)   (1/0)   (1/1)  rate(%) priority  
rate
  ----- --- ----- -------- -------- ---- ------- ------- ------- ------- ------- ------
-- -------
   147  0   95     95      0     off     1      1      1      1      0      1   -N.A-
       3    5     5      0     off     1      1      1      1      0      1   -N.A-
   Schd  Shp-type      PIR       CIR       DBR      Exc-rate
   1(a)   0        0   1000000000   1000000000  Unspec

   Node-Type       EXC_BW   Internal-Node  Flags  Node flags
     unknown      32640000         Level3 0x00001   0x0000000

   Exc-burst   Guar-burst
        0          0
Bandwidth : 1000000000
Service profile params : 0
L2 Scheduler          : 0
L3 Scheduler          : 1
Scheduler Level        : 1
Default L3 scheduler   : yes

Input scheduling policy
========================

  - No scheduling policy attached -

ADPC3(Weir-re1 vty)# exit
```

Notice that it only shows queue numbers 0 and 3, which have the default 95% and 0% transmit-rate values, respectively. It tells us that the configured scheduler-map "CORE" is not programmed and assigned to the interface ge-3/0/0 at the PFE hardware level. That also explains why the RED drops were occurring in the EF and AF queues, instead of in the lower priority BE queue. The BE queue has the default configuration, 95% of the transmit-rate and buffer, while the AF and EF queues have 0%.

Since the routing-engine has the "CORE" scheduler-map configuration assigned to the ge-3/0/0, but it's not programmed in the PFE, let's check what happens when the scheduler-map is first applied to the interface and committed. Then look back at the syslog messages at the time in which this scheduler-map was applied to the interface and committed, to see if there were any errors. The following errors were reported in the syslog messages.

```
{master}[edit]
lab@Weir-re0# run show log messages
<truncated>
Mar  4 12:01:02.425  Weir-re0 l2cp[2260]: Read acess profile () config
Mar  4 12:01:02.457  Weir-re0 idpd[1450]: IDP_COMMIT_COMPLETED: IDP policy commit is
complete.
Mar  4 12:01:09.244  Weir-re0 xntpd[2240]: kernel time sync disabled 2041
Mar  4 12:01:11.244  Weir-re0 xntpd[2240]: kernel time sync enabled 2001
Mar  4 12:29:44.844  Weir-re0 mgd[4655]: UI_COMMIT: User 'lab' requested 'commit'
operation (comment: none)
Mar  4 12:29:46.982  Weir-re0 ffp[7810]: "dynamic-profiles": No change to profiles
Mar  4 12:29:47.402  Weir-re0 mgd[4655]: UI_LOAD_EVENT: User 'lab' is performing a
'rollback 1'
Mar  4 12:29:47.661  Weir-re0 mgd[4655]: UI_COMMIT: User 'lab' requested 'commit'
operation (comment: none)
Mar  4 12:29:49.731  Weir-re0 ffp[7841]: "dynamic-profiles": No change to profiles
Mar  4 12:29:49.888  Weir-re0 cosd[2246]: COSD_TX_QUEUE_RATES_TOO_HIGH: Unable to apply
scheduler map CORE to interface ge-3/0/0: sum of scheduler transmission rates exceeds
interface shaping or transmission rate
```

The highlighted error message above, reported by the class-of-service daemon, indicates that the system was unable to apply the “CORE” scheduler-map to the interface ge-3/0/0 because the sum of the configured transmission-rate values in each associated scheduler adds up to over 100%.

Let’s take a closer look at the scheduler-map configuration. Notice the highlighted transmission-rate values on the associated schedulers:

```
scheduler-maps {
    CORE {
        forwarding-class BE scheduler CORE-BE;
        forwarding-class AF scheduler CORE-AF;
        forwarding-class EF scheduler CORE-EF;
        forwarding-class NC scheduler CORE-NC;
    }
}
schedulers {
    CORE-BE {
        transmit-rate 765m;
        buffer-size {
            remainder;
        }
        priority medium-low;
    }
    CORE-AF {
        transmit-rate percent 18;
        buffer-size percent 18;
        priority medium-high;
    }
    CORE-EF {
        transmit-rate 135m;
        buffer-size percent 10;
        priority high;
    }
    CORE-NC {
        transmit-rate percent 5;
        buffer-size percent 5;
        priority strict-high;
    }
}
```

Some of the schedulers' transmit-rate values are using a bandwidth value while the others have a percentage value. For our gigabit-ethernet link, the sum of the transmit-rate values here add up to over 100% of the link!

When configuring class-of-service scheduler-maps, the sum of the associated schedulers must add up to 100%, otherwise it is considered as an invalid configuration that will not be programmed at the PFE hardware level. As a result, scenarios similar to the problem described in this recipe will occur.

NOTE When configuring scheduler-maps, it's a good idea to consider configuring the transmit-rate of the associated schedulers in either all transmit-rate percentage or bandwidth values, instead of a mixture of both. This helps avoid confusion and makes it less likely to run into this type of problem.

To fix the issue, the EF queue is assigned a transmit-rate of 15 percent while the BE queue is assigned the remainder. So now, the sum of the transmit-rates in the associated schedulers adds up to 100%:

```
{master}[edit]
lab@Weir-re0# set class-of-service schedulers CORE-EF transmit-rate percent 15

{master}[edit]
lab@Weir-re0# set class-of-service schedulers CORE-BE transmit-rate remainder

{master}[edit]
lab@Weir-re0# show class-of-service schedulers
CORE-BE {
    transmit-rate remainder;
    buffer-size {
        remainder;
    }
    priority medium-low;
}
CORE-AF {
    transmit-rate percent 18;
    buffer-size percent 18;
    priority medium-high;
}
CORE-EF {
    transmit-rate percent 15;
    buffer-size percent 10;
    priority high;
}
CORE-NC {
    transmit-rate percent 5;
    buffer-size percent 5;
    priority strict-high;
}

{master}[edit]
lab@Weir-re0# commit
re0:
configuration check succeeds
re1:
commit complete
re0:
commit complete
```

Just to make sure that there were no class-of-service related errors in the syslog after the commit, let's check the messages again:

```
{master}[edit]
lab@Weir-re0# run show log messages
<truncated>
Mar  4 12:32:37.485  Weir-re0 mgd[4655]: UI_COMMIT: User 'lab' requested 'commit'
operation (comment: none)
Mar  4 12:32:39.620  Weir-re0 ffp[7880]: "dynamic-profiles": No change to profiles
```

Next, the interface statistics were cleared then checked for packet drops. Now, the drops are happening in the lower priority BE queue as expected, instead of the higher priority AF or EF queues:

```
{master}[edit]
lab@Weir-re0# run clear interfaces statistics all

{master}[edit]
lab@Weir-re0# run show interfaces ge-3/0/0 extensive
Physical interface: ge-3/0/0, Enabled, Physical link is Up
  Interface index: 147, SNMP ifIndex: 1546, Generation: 395
  Link-level type: Ethernet, MTU: 1514, Speed: 1000mbps, BPDU Error: None, MAC-
REWRITE Error: None,
  Loopback: Disabled, Source filtering: Disabled, Flow control: Enabled, Auto-
negotiation: Enabled,
  Remote fault: Online
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x4000
  Link flags     : None
  CoS queues     : 8 supported, 4 maximum usable queues
  Schedulers     : 0
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: 78:19:f7:50:23:de, Hardware address: 78:19:f7:50:23:de
  Last flapped   : 2012-03-04 10:36:34 PST (01:57:57 ago)
  Statistics last cleared: 2012-03-04 12:34:16 PST (00:00:15 ago)
  Traffic statistics:
   Input  bytes  :                  160                    0 bps
   Output bytes  :           1946288304           1055032760 bps
   Input  packets:                    2                    0 pps
   Output packets:              2517406               170671 pps
```

```
    IPv6 transit statistics:
     Input  bytes  :                 0
     Output bytes  :                 0
     Input  packets:                 0
     Output packets:                 0
   Dropped traffic statistics due to STP State:
    Input  bytes  :                  0
    Output bytes  :                  0
    Input  packets:                  0
    Output packets:                  0
   Input errors:
    Errors: 0, Drops: 0, Framing errors: 0, Runts: 0, Policed discards: 0, L3 
incompletes: 0, L2 channel errors: 0,
    L2 mismatch timeouts: 0, FIFO errors: 0, Resource errors: 0
  Output errors:
    Carrier transitions: 0, Errors: 0, Drops: 0, Collisions: 0, Aged packets: 0, FIFO 
errors: 0,
    HS link CRC errors: 0, MTU errors: 0, Resource errors: 0
  Egress queues: 8 supported, 4 in use
  Queue counters:       Queued packets  Transmitted packets      Dropped packets
    0 BE                       1813224              1523921               289316
    1 AF                        426707               426660                    0
    2 EF                        320011               320009                    0
    3 NC                             2                    2                    0
```

Let's run the `show interfaces queue` command to double-check that that the issue is not happening anymore:

```
{master}[edit]
lab@Weir-re0# run show interfaces queue ge-3/0/0
Physical interface: ge-3/0/0, Enabled, Physical link is Up
  Interface index: 147, SNMP ifIndex: 1546
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Queue: 0, Forwarding classes: BE
  Queued:
    Packets              :               4110287                120706 pps
    Bytes                :            3251282164             764972816 bps
  Transmitted:
    Packets              :               3454399                 101439 pps
    Bytes                :            2808172131             660610696 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                655893                 19261 pps
     Low                 :                655893                 19261 pps
     Medium-low          :                     0                     0 pps
     Medium-high         :                     0                     0 pps
     High                :                     0                     0 pps
    RED-dropped bytes    :             443112593             104337544 bps
     Low                 :             443112593             104337544 bps
     Medium-low          :                     0                     0 bps
     Medium-high         :                     0                     0 bps
     High                :                     0                     0 bps
Queue: 1, Forwarding classes: AF
  Queued:
    Packets              :                967157                 28459 pps
    Bytes                :            764837424            180142248 bps
  Transmitted:
    Packets              :               967123                28437 pps
    Bytes                :            764820016            180052136 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                    0                    0 pps
     Low                 :                    0                    0 pps
     Medium-low          :                    0                    0 pps
     Medium-high         :                    0                    0 pps
     High                :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
     Low                 :                    0                    0 bps
     Medium-low          :                    0                    0 bps
     Medium-high         :                    0                    0 bps
     High                :                    0                    0 bps
Queue: 2, Forwarding classes: EF
  Queued:
    Packets              :               725354                21334 pps
    Bytes                :            574096613            135153520 bps
  Transmitted:
    Packets              :               725354                21334 pps
    Bytes                :            574096613            135153520 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                    0                    0 pps
     Low                 :                    0                    0 pps
     Medium-low          :                    0                    0 pps
     Medium-high         :                    0                    0 pps
     High                :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
     Low                 :                    0                    0 bps
     Medium-low          :                    0                    0 bps
     Medium-high         :                    0                    0 bps
     High                :                    0                    0 bps
Queue: 3, Forwarding classes: NC
  Queued:
    Packets              :                    4                    0 pps
    Bytes                :                  392                    0 bps
  Transmitted:
    Packets              :                    4                     0 pps
    Bytes                :                  392                    0 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                    0                    0 pps
     Low                 :                    0                    0 pps
     Medium-low          :                    0                    0 pps
     Medium-high         :                    0                    0 pps
     High                :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
     Low                 :                    0                    0 bps
     Medium-low          :                    0                    0 bps
     Medium-high         :                    0                    0 bps
     High                :                    0                    0 bps
```

And, as expected, RED drops are only present in the lower priority BE queue. Just out of curiosity, let's check the PFE programming for this interface to make sure the changes are reflected there. First, log in to the PFE. You should now know that using shell commands is not supported or endorsed by Juniper Networks, but let's proceed like careful, prudent engineers who want to solve a problem:

```
{master}[edit]
lab@Weir-re0# run start shell pfe network fpc3

ADPC platform (1200Mhz MPC 8548 processor, 1024MB memory, 512KB flash)

ADPC3(Weir-re1 vty)#
```

Next, verify the IFD index number for the interface:

```
ADPC3(Weir-re1 vty)# show ifd brief
Index  Name                  Type         Flags  Slot   State
-----  --------------------  -----------  ------ -----  ------
<truncated>
  147  ge-3/0/0              Ethernet     0x0000000000008000     3  Up
<truncated>
```

The IFD index number is 147, still for the ge-3/0/0 interface.

Now, let's check the class-of-service scheduler-map programming for the ge-3/0/0 interface:

```
ADPC3(Weir-re1 vty)# show f-cos scheduling-policy 147
Output scheduling policy
=======================
ifd ge-3/0/0 (147)
   ifd   Q  tx-BW delay-BW  sched   rate tcp/plp tcp/plp tcp/plp tcp/plp excess  excess
shaping
  index num  (%)    (%)  priority cntl  (0/0)   (0/1)   (1/0)   (1/1)  rate(%) priority
rate
  ----- --- ----- ------ ------ ---- ----- ----- ----- ----- ----- ------ -------
   147    0   255    255     1    off    1     1     1     1     0     0    -N.A-
          1    18     18     2    off    1     1     1     1     0     0    -N.A-
          2    15     10     3    off    1     1     1     1     0     0    -N.A-
          3     5      5     4    off    1     1     1     1     0     0    -N.A-
   Schd   Shp-type       PIR        CIR          DBR        Exc-rate
    304(a)    0          0     1000000000    1000000000   Unspec

    Node-Type         EXC_BW    Internal-Node   Flags  Node flags
      unknown       32640000           Level3 0x00000   0x0000000

    Exc-burst   Guar-burst  t
            0           0
Bandwidth : 1000000000
Service profile params : 2257
L2 Scheduler           : 0
L3 Scheduler           : 304
Scheduler Level        : 1
Default L3 scheduler   : yes

Input scheduling policy
=======================

  - No scheduling policy attached -
```

Notice all four queues this time, where previously only the default class-of service settings were applied.

Summary

The router was observed to be dropping high priority traffic rather than the lower priority best effort traffic. In real world networks, this could have a big impact on the business.

It's important to understand the troubleshooting involved with identifying the cause of the problem so the problem can be quickly fixed. Important CLI commands to check for packet drops include `show interfaces extensive <interface-name>` and `show interfaces queue <interface-name>`. These commands will show you which queue the packets are dropping in. Run the commands a few times in sucsession so you can tell if the counters in the outputs are incrementing. We noticed packet drops in both the AF and EF queues, but not the lower priority BE queue.

Next, the scheduler map configuration was checked and we could see that the scheduler map was applied to the egress interface as expected. There were no obvious problems.

So we decided to look at the syslog messages and the time in which the scheduler map was first configured and applied to the interface. Oftentimes, the syslog is a great tool to use when dealing with class of service configurations because the class of service daemon usually writes relevant error messages there. In looking at the syslog, we noticed an error message, which indicated that the scheduler map transmit rate values exceeded 100%, and thus it could not program such configuration to the PFE. As a result, the egress interface did not have the scheduler map applied at the PFE level. We logged into the PFE to verify that this was in fact the case.

After taking a closer look at the configuration of the associated schedulers within the scheduler map, we identified the problem. We had some schedulers configured to use a transmit-rate value as a hard coded bandwidth value, and some schedulers used a transmit-rate value as a percent of the link bandwidth value. Usually it's a good idea to configure the schedulers to contain either all bandwidth percent values or hard-coded bandwidth values within the scheduler map, but not a mixture of both – it makes reading the configuration much easier and lessens the risk of running into this type of configuration problem.

Recipe 8

Shaping Rate Dropping Packets

This recipe shows us how to identify that packet drops are being reported and the troubleshooting steps necessary to find out what is causing them. It also shows us how packet headers are accounted for in certain statistics in Junos.

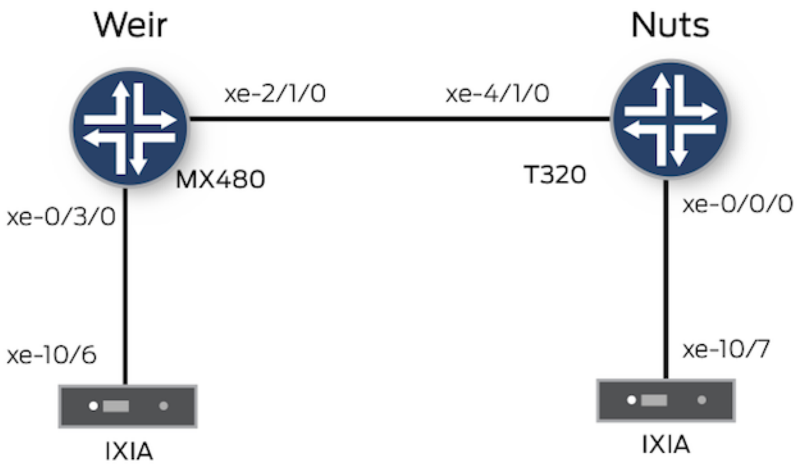
Problem

The router is dropping packets, even though the traffic rate being sent through the interface is under the configured class-of-service shaping-rate value. Engineers would not expect to see any packet drops and this may lead them to think that there is an issue with the class of service configuration or a software problem.

This behavior could occur due to a configuration mistake on the router or even on traffic generator equipment, a software problem, or a hardware problem. Let's investigate.

Solution

Our lab routers, Weir and Nuts, are MX480 and T320, respectively. They are both running Junos 11.4R1.14, and they are connected to an IXIA Traffic Generator.



The following IXIA traffic streams are sent from IXIA port 10/6 to port 10/7:

Stream 1

```
Stream #1 Best Effort
Source IP - 192.168.1.2
Destination IP - 192.168.2.2
Protocol – UDP
Frame Size – 64 bytes
TOS – DSCP 000000 which represents BE
Rate – 500Mbps
```

Stream 2

```
Stream #2 Assured Forwarding
Source IP - 192.168.1.2
Destination IP - 192.168.2.2
Protocol – UDP
Frame Size – 64 bytes
TOS – DSCP 001010 which represents AF
Rate – 100Mbps
```

Stream 3

```
Stream #3 Expedited Forwarding
Source IP - 192.168.1.2
Destination IP - 192.168.2.2
Protocol – UDP
Frame Size – 64 bytes
TOS – DSCP 101110 which represents EF
Rate – 300Mbps
```

The DSCP bit markings are different for each of the three traffic streams, which allows the routers to classify these packets and forward them out of three different queues. For example, Stream 1 uses the Best Effort (BE) queue, Stream 2 uses the Assured Forwarding (AF) queue, and Stream 3 uses the Expedited Forwarding (EF) queue.

Also, notice the different traffic rates that are configured on the IXIA Traffic Generator. Stream 1 is sending 500Mbps of traffic, Stream 2 is sending 100Mbps, and Stream 3 is sending 300Mbps, adding up to about 900Mbps of traffic that the IXIA is sending from port 10/6 to port 10/7. These are all 10GE links in the topology, so there is more than enough bandwidth available to handle 900Mbps. The Weir xe-2/1/0 link has a shaping-rate configured for 1Gbps as shown here:

```
{master}[edit]
lab@Weir-re0# show class-of-service interfaces
      <...>
      xe-2/1/0 {
         scheduler-map CORE;
         shaping-rate 1g;
         unit 0 {
            classifiers {
               dscp CLASSIFIER_1;
            }
         }
      }
      <...>
```

And here is the traffic coming into the Weir router, on xe-0/3/0, from the IXIA:

```
{master}[edit]
lab@Weir-re0# run show interfaces xe-0/3/0 extensive
Physical interface: xe-0/3/0, Enabled, Physical link is Up
  Interface index: 243, SNMP ifIndex: 1296, Generation: 491
  Link-level type: Ethernet, MTU: 1514, LAN-
PHY mode, Speed: 10Gbps, Loopback: None, Source filtering: Disabled,
  Flow control: Enabled
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x4000
  Link flags     : None
  CoS queues     : 8 supported, 8 maximum usable queues
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: 78:19:f7:50:20:f6, Hardware address: 78:19:f7:50:20:f6
  Last flapped   : 2012-03-03 11:22:58 PST (1d 03:29 ago)
  Statistics last cleared: 2012-03-04 14:52:35 PST (00:00:15 ago)
  Traffic statistics:
   Input  bytes  :          1286250942           646872496 bps
   Output bytes  :                   0                   0 bps
   Input  packets:            27961977             1757805 pps
   Output packets:                   0                   0 pps
<...>
```

Notice the input bps value here, which is about 646Mbps, even though the IXIA streams are configured to send 900Mbps. The reason for the discrepancy is because of the way Junos accounts traffic statistics in the `show interfaces extensive` command output. Junos does not include the header information of the IXIA frames, instead it includes only the data portion of the frames.

Next, let's check the egress traffic on the Weir router – it's xe-2/1/0 interface has the 1Gbps class-of-service shaping-rate configured:

```
{master}[edit]
lab@Weir-re0# run show interfaces xe-2/1/0 extensive
Physical interface: xe-2/1/0, Enabled, Physical link is Up
  Interface index: 145, SNMP ifIndex: 1537, Generation: 393
  Link-level type: Ethernet, MTU: 1514, LAN-PHY mode, Speed: 10Gbps, Loopback: None,
Source filtering: Disabled,
  Flow control: Enabled
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x4000
  Link flags     : None
  CoS queues     : 8 supported, 8 maximum usable queues
  Hold-times      : Up 0 ms, Down 0 ms
   Current address: 78:19:f7:50:22:e6, Hardware address: 78:19:f7:50:22:e6
   Last flapped   : 2012-03-03 11:21:04 PST (1d 03:31 ago)
   Statistics last cleared: 2012-03-04 14:52:35 PST (00:00:26 ago)
   Traffic statistics:
    Input  bytes  :                   240                  368 bps
    Output bytes  :            1758169124            547612440 bps
    Input  packets:                     3                    0 pps
    Output packets:              38221064              1488077 pps
    IPv6 transit statistics:
     Input  bytes  :                    0
     Output bytes  :                    0
     Input  packets:                    0
     Output packets:                    0
   Label-switched interface (LSI) traffic statistics:
    Input  bytes  :                     0                    0 bps
    Input  packets:                     0                    0 pps
   Dropped traffic statistics due to STP State:
    Input  bytes  :                     0
    Output bytes  :                     0
    Input  packets:                     0
    Output packets:                     0
  Input errors:
    Errors: 0, Drops: 0, Framing errors: 0, Runts: 0, Policed discards: 0, L3
incompletes: 0, L2 channel errors: 0,
    L2 mismatch timeouts: 0, FIFO errors: 0, Resource errors: 0
  Output errors:
    Carrier transitions: 0, Errors: 0, Drops: 6797986, Collisions: 0, Aged packets: 0,
FIFO errors: 0,
     HS link CRC errors: 0, MTU errors: 0, Resource errors: 0
   Egress queues: 8 supported, 4 in use
   Queue counters:       Queued packets  Transmitted packets      Dropped packets
```

```
    0 BE                   24609657             22499628              2110016
    1 EF                   14765797             11252657              3513132
    2 AF                    4921931              3747091              1174838
    3 NC                          3                    3                    0
<...>
```

Notice that the egress traffic rate is lower than the received traffic rate, and that there are some queue drops. Let's run the egress xe-2/1/0 interface statistics again to check if the drops are incrementing:

```
{master}[edit]
lab@Weir-re0# run show interfaces xe-2/1/0 extensive
Physical interface: xe-2/1/0, Enabled, Physical link is Up
  Interface index: 145, SNMP ifIndex: 1537, Generation: 393
  Link-level type: Ethernet, MTU: 1514, LAN-
PHY mode, Speed: 10Gbps, Loopback: None, Source filtering: Disabled,
  Flow control: Enabled
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x4000
  Link flags     : None
  CoS queues     : 8 supported, 8 maximum usable queues
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: 78:19:f7:50:22:e6, Hardware address: 78:19:f7:50:22:e6
  Last flapped   : 2012-03-03 11:21:04 PST (1d 03:32 ago)
  Statistics last cleared: 2012-03-04 14:52:35 PST (00:00:39 ago)
  Traffic statistics:
   Input  bytes  :                  320                    0 bps
   Output bytes  :           2616547538            547608376 bps
   Input  packets:                    4                    0 pps
   Output packets:             56881463              1488066 pps
   IPv6 transit statistics:
    Input  bytes  :                  0
    Output bytes  :                  0
    Input  packets:                  0
    Output packets:                  0
  Label-switched interface (LSI) traffic statistics:
   Input  bytes  :                    0                    0 bps
   Input  packets:                    0                    0 pps
  Dropped traffic statistics due to STP State:
   Input  bytes  :                    0
   Output bytes  :                    0
   Input  packets:                    0
   Output packets:                    0
  Input errors:
    Errors: 0, Drops: 0, Framing errors: 0, Runts: 0, Policed discards: 0, L3 incompletes:
0, L2 channel errors: 0,
    L2 mismatch timeouts: 0, FIFO errors: 0, Resource errors: 0
  Output errors:
    Carrier transitions: 0, Errors: 0, Drops: 10682537, Collisions: 0, Aged packets: 0,
FIFO errors: 0,
     HS link CRC errors: 0, MTU errors: 0, Resource errors: 0
  Egress queues: 8 supported, 4 in use
  Queue counters:       Queued packets  Transmitted packets      Dropped packets
    0 BE                   38672272             35356510              3315735
    1 EF                      23203367            17682749            5520629
    2 AF                       7734454             5888278            1846173
    3 NC                             4                   4                  0
<...>
```

Notice that the egress drops counter and dropped packets for each queue are incrementing.

Another useful CLI command to check for packet drops is the `show interfaces queue` command, which can even show the RED-dropped packet rate (Random Early Drop) for each queue instead of just a counter:

```
{master}[edit]
lab@Weir-re0# run show interfaces queue xe-2/1/0
Physical interface: xe-2/1/0, Enabled, Physical link is Up
  Interface index: 145, SNMP ifIndex: 1537
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Queue: 0, Forwarding classes: BE
  Queued:
    Packets              :              45703605                976575 pps
    Bytes                :            2102365830             359379800 bps
  Transmitted:
    Packets              :              41784975                892842 pps
    Bytes                :            1922108850             328565936 bps
    Tail-dropped packets :                     0                     0 pps
    RED-dropped packets  :               3918597                 83730 pps
     Low                 :               3918597                 83730 pps
     Medium-low          :                     0                     0 pps
     Medium-high         :                     0                     0 pps
     High                :                     0                     0 pps
    RED-dropped bytes    :             180255462              30812944 bps
     Low                 :             180255462              30812944 bps
     Medium-low          :                     0                     0 bps
     Medium-high         :                     0                     0 bps
     High                :                     0                     0 bps
Queue: 1, Forwarding classes: EF
  Queued:
    Packets              :              27422167                585945 pps
    Bytes                :            1261419682             215627856 bps
  Transmitted:
    Packets              :              20897793                446534 pps
    Bytes                :             961298478             164324672 bps
    Tail-dropped packets :                     0                     0 pps
    RED-dropped packets  :               6524381                139410 pps
     Low                 :                     0                     0 pps
     Medium-low          :                     0                     0 pps
     Medium-high         :                     0                     0 pps
     High                :               6524381                139410 pps
    RED-dropped bytes    :             300121526              51302880 bps
     Low                 :                     0                     0 bps
     Medium-low          :                     0                     0 bps
     Medium-high         :                     0                     0 bps
     High                 :           300121526             51302880 bps
Queue: 2, Forwarding classes: AF
  Queued:
    Packets               :             9140721               195315 pps
    Bytes                 :           420473166             71876016 bps
  Transmitted:
    Packets               :             6958876               148694 pps
    Bytes                 :           320108296             54719448 bps
    Tail-dropped packets :                    0                    0 pps
    RED-dropped packets   :             2181841                46620 pps
     Low                  :             2181841                46620 pps
     Medium-low           :                   0                    0 pps
     Medium-high          :                   0                    0 pps
     High                 :                   0                    0 pps
    RED-dropped bytes     :           100364686             17156464 bps
     Low                  :           100364686             17156464 bps
     Medium-low           :                   0                    0 bps
     Medium-high          :                   0                    0 bps
     High                 :                   0                    0 bps
Queue: 3, Forwarding classes: NC
  Queued:
    Packets               :                   5                    0 pps
    Bytes                 :                 530                    0 bps
  Transmitted:
    Packets               :                   5                     0 pps
    Bytes                 :                 530                    0 bps
    Tail-dropped packets :                    0                    0 pps
    RED-dropped packets   :                   0                    0 pps
     Low                  :                   0                    0 pps
     Medium-low           :                   0                    0 pps
     Medium-high          :                   0                    0 pps
     High                 :                   0                    0 pps
    RED-dropped bytes     :                   0                    0 bps
     Low                  :                   0                    0 bps
     Medium-low           :                   0                    0 bps
     Medium-high          :                   0                    0 bps
     High                 :                   0                    0 bps
```

And this confirms that the router is dropping packets, even though the traffic rate appears to be under the configured class-of-service shaping-rate value of 1G.

So, let’s try to increase the shaping-rate to 1100M to see if that helps prevent the packet drops:

```
{master}[edit]
lab@Weir-re0# show class-of-service interfaces xe-2/1/0
scheduler-map CORE;
shaping-rate 1100000000;
unit 0 {
    classifiers {
        dscp CLASSIFIER_1;
    }
}
```

After making the shaping-rate change to 1100M, let's clear the interface statistics, then check again:

```
{master}[edit]
lab@Weir-re0# run clear interfaces statistics all

{master}[edit]
lab@Weir-re0# run show interfaces xe-0/3/0 extensive
Physical interface: xe-0/3/0, Enabled, Physical link is Up
  Interface index: 243, SNMP ifIndex: 1296, Generation: 491
  Link-level type: Ethernet, MTU: 1514, LAN-
PHY mode, Speed: 10Gbps, Loopback: None, Source filtering: Disabled,
  Flow control: Enabled
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x4000
  Link flags     : None
  CoS queues     : 8 supported, 8 maximum usable queues
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: 78:19:f7:50:20:f6, Hardware address: 78:19:f7:50:20:f6
  Last flapped   : 2012-03-03 11:22:58 PST (1d 03:52 ago)
  Statistics last cleared: 2012-03-04 15:15:20 PST (00:00:13 ago)
  Traffic statistics:
   Input  bytes  :           1125469580            646875240 bps
   Output bytes  :                    0                    0 bps
   Input  packets:             24466730              1757813 pps
   Output packets:                    0                    0 pps
<...>
```

Now check the egress xe-2/1/0 statistics:

```
{master}[edit]
lab@Weir-re0# run show interfaces xe-2/1/0 extensive
Physical interface: xe-2/1/0, Enabled, Physical link is Up
  Interface index: 145, SNMP ifIndex: 1537, Generation: 393
  Link-level type: Ethernet, MTU: 1514, LAN-
PHY mode, Speed: 10Gbps, Loopback: None, Source filtering: Disabled,
  Flow control: Enabled
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x4000
  Link flags     : None
  CoS queues     : 8 supported, 8 maximum usable queues
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: 78:19:f7:50:22:e6, Hardware address: 78:19:f7:50:22:e6
  Last flapped   : 2012-03-03 11:21:04 PST (1d 03:54 ago)
  Statistics last cleared: 2012-03-04 15:15:20 PST (00:00:20 ago)
  Traffic statistics:
   Input  bytes  :                  160                    0 bps
   Output bytes  :           1522909254            602331488 bps
   Input  packets:                    2                    0 pps
   Output packets:             33106719              1636769 pps
   IPv6 transit statistics:
    Input  bytes  :                   0
    Output bytes  :                   0
    Input  packets:                   0
    Output packets:                   0
  Label-switched interface (LSI) traffic statistics:
   Input  bytes  :                    0                   0 bps
   Input  packets:                    0                   0 pps
  Dropped traffic statistics due to STP State:
   Input  bytes  :                    0
   Output bytes  :                    0
   Input  packets:                    0
   Output packets:                    0
  Input errors:
    Errors: 0, Drops: 0, Framing errors: 0, Runts: 0, Policed discards: 0, L3
incompletes: 0, L2 channel errors: 0,
    L2 mismatch timeouts: 0, FIFO errors: 0, Resource errors: 0
 Output errors:
    Carrier transitions: 0, Errors: 0, Drops: 2615051, Collisions: 0, Aged packets: 0,
FIFO errors: 0,
    HS link CRC errors: 0, MTU errors: 0, Resource errors: 0
  Egress queues: 8 supported, 4 in use
  Queue counters:       Queued packets  Transmitted packets      Dropped packets
    0 BE                      21093923             21093932                    0
    1 EF                      12656357             10726157              1930204
    2 AF                       4218785              3533939               684847
    3 NC                             3                    3                    0
<...>
```

There are still drops reported.

Let's check the xe-2/1/0 interface statistics again to confirm that the drops are incrementing:

```
{master}[edit]
lab@Weir-re0# run show interfaces xe-2/1/0 extensive
Physical interface: xe-2/1/0, Enabled, Physical link is Up
  Interface index: 145, SNMP ifIndex: 1537, Generation: 393
  Link-level type: Ethernet, MTU: 1514, LAN-
PHY mode, Speed: 10Gbps, Loopback: None, Source filtering: Disabled,
  Flow control: Enabled
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x4000
  Link flags     : None
  CoS queues     : 8 supported, 8 maximum usable queues
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: 78:19:f7:50:22:e6, Hardware address: 78:19:f7:50:22:e6
  Last flapped   : 2012-03-03 11:21:04 PST (1d 03:55 ago)
  Statistics last cleared: 2012-03-04 15:15:20 PST (00:00:47 ago)
  Traffic statistics:
   Input  bytes  :                  400                   0 bps
   Output bytes  :           3561272428           602300416 bps
   Input  packets:                    5                   0 pps
   Output packets:             77418958             1636685 pps
   IPv6 transit statistics:
    Input  bytes  :                   0
    Output bytes  :                   0
    Input  packets:                   0
    Output packets:                   0
  Label-switched interface (LSI) traffic statistics:
   Input  bytes  :                    0                    0 bps
   Input  packets:                    0                    0 pps
  Dropped traffic statistics due to STP State:
   Input  bytes  :                    0
   Output bytes  :                    0
   Input  packets:                    0
   Output packets:                    0
  Input errors:
    Errors: 0, Drops: 0, Framing errors: 0, Runts: 0, Policed discards: 0, L3
incompletes: 0, L2 channel errors: 0,
    L2 mismatch timeouts: 0, FIFO errors: 0, Resource errors: 0
  Output errors:
    Carrier transitions: 0, Errors: 0, Drops: 5665946, Collisions: 0, Aged packets: 0,
FIFO errors: 0,
    HS link CRC errors: 0, MTU errors: 0, Resource errors: 0
  Egress queues: 8 supported, 4 in use
  Queue counters:       Queued packets  Transmitted packets      Dropped packets
    0 BE                      45703550             45703528                    0
    1 EF                      27422136             23240038              4182109
    2 AF                       9140711              7656869              1483837
    3 NC                             5                    5                    0
```

And they are. So let's use the `show interfaces queue` command on this xe-2/1/0 link to look at the RED dropped packet rates:

```
{master}[edit]
lab@Weir-re0# run show interfaces queue xe-2/1/0
Physical interface: xe-2/1/0, Enabled, Physical link is Up
  Interface index: 145, SNMP ifIndex: 1537
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Queue: 0, Forwarding classes: BE
  Queued:
    Packets              :              56250540                975636 pps
    Bytes                :            2587524840             359034408 bps
  Transmitted:
    Packets              :              56250520                 975642 pps
    Bytes                :            2587523920             359036256 bps
    Tail-dropped packets :                     0                     0 pps
    RED-dropped packets  :                     0                     0 pps
     Low                 :                     0                     0 pps
     Medium-low          :                     0                     0 pps
     Medium-high         :                     0                     0 pps
     High                :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
     Low                 :                     0                     0 bps
     Medium-low          :                     0                     0 bps
     Medium-high         :                     0                     0 bps
     High                :                     0                     0 bps
Queue: 1, Forwarding classes: EF
  Queued:
    Packets              :              33750330                585382 pps
    Bytes                :            1552515180             215420648 bps
  Transmitted:
    Packets              :             28603129                496109 pps
    Bytes                :           1315743934             182568120 bps
    Tail-dropped packets :                    0                     0 pps
    RED-dropped packets  :              5147215                 89277 pps
     Low                 :                    0                     0 pps
     Medium-low          :                    0                     0 pps
     Medium-high         :                    0                     0 pps
     High                :              5147215                 89277 pps
    RED-dropped bytes    :            236771890              32853952 bps
     Low                 :                    0                     0 bps
     Medium-low          :                    0                     0 bps
     Medium-high         :                    0                     0 bps
     High                :            236771890              32853952 bps
Queue: 2, Forwarding classes: AF
  Queued:
    Packets              :             11250109                195127 pps
    Bytes                :            517505014              71806880 bps
  Transmitted:
    Packets              :              9423842                163450 pps
    Bytes                :            433496732              60149824 bps
    Tail-dropped packets :                    0                     0 pps
    RED-dropped packets  :              1826261                 31675 pps
     Low                 :              1826261                 31675 pps
     Medium-low          :                    0                     0 pps
     Medium-high         :                    0                     0 pps
     High                :                    0                     0 pps
    RED-dropped bytes    :             84008006              11656432 bps
     Low                 :             84008006              11656432 bps
     Medium-low          :                    0                     0 bps
     Medium-high         :                    0                     0 bps
     High                :                    0                     0 bps
Queue: 3, Forwarding classes: NC
  Queued:
    Packets              :                    7                     0 pps
    Bytes                :                  742                   232 bps
  Transmitted:
    Packets              :                    7                      0 pps
    Bytes                :                  742                   232 bps
    Tail-dropped packets :                    0                     0 pps
    RED-dropped packets  :                    0                     0 pps
     Low                 :                    0                     0 pps
     Medium-low          :                    0                     0 pps
     Medium-high         :                    0                     0 pps
     High                :                    0                     0 pps
    RED-dropped bytes    :                    0                     0 bps
     Low                 :                    0                     0 bps
     Medium-low          :                    0                     0 bps
     Medium-high         :                    0                     0 bps
     High                :                    0                     0 bps
```

Compared to the first test when shaping-rate 1G was configured, there are not as many drops happening. Increasing the shaping-rate from 1G to 1100M helped a little, because the drops are at a lower rate, but it didn't eliminate the problem.

So let's change the shaping-rate from 1100M to 1200M to see if that helps clear the problem:

```
{master}[edit]
lab@Weir-re0# show class-of-service interfaces xe-2/1/0
scheduler-map CORE;
shaping-rate 1200000000;
unit 0 {
    classifiers {
        dscp CLASSIFIER_1;
    }
}
```

First let's check the ingress interface to see what the packet rate is coming into the Weir router from IXIA:

```
{master}[edit]
lab@Weir-re0# run clear interfaces statistics all

{master}[edit]
lab@Weir-re0# run show interfaces xe-0/3/0 extensive
Physical interface: xe-0/3/0, Enabled, Physical link is Up
  Interface index: 243, SNMP ifIndex: 1296, Generation: 491
  Link-level type: Ethernet, MTU: 1514, LAN-PHY mode, Speed: 10Gbps, Loopback: None,
Source filtering: Disabled,
  Flow control: Enabled
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x4000
  Link flags     : None
  CoS queues     : 8 supported, 8 maximum usable queues
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: 78:19:f7:50:20:f6, Hardware address: 78:19:f7:50:20:f6
  Last flapped   : 2012-03-03 11:22:58 PST (1d 03:55 ago)
  Statistics last cleared: 2012-03-04 15:18:24 PST (00:00:07 ago)
  Traffic statistics:
   Input  bytes  :            643125448            646872776 bps
   Output bytes  :                    0                    0 bps
   Input  packets:             13980988              1757806 pps
   Output packets:                    0                    0 pps
<...>
```

Now check the egress xe-2/1/0 interface statistics to see if there are any more packet drops:

```
{master}[edit]
lab@Weir-re0# run show interfaces xe-2/1/0 extensive
Physical interface: xe-2/1/0, Enabled, Physical link is Up
  Interface index: 145, SNMP ifIndex: 1537, Generation: 393
  Link-level type: Ethernet, MTU: 1514, LAN-PHY mode, Speed: 10Gbps, Loopback: None,
Source filtering: Disabled,
  Flow control: Enabled
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x4000
  Link flags     : None
  CoS queues     : 8 supported, 8 maximum usable queues
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: 78:19:f7:50:22:e6, Hardware address: 78:19:f7:50:22:e6
  Last flapped   : 2012-03-03 11:21:04 PST (1d 03:57 ago)
  Statistics last cleared: 2012-03-04 15:18:24 PST (00:00:13 ago)
  Traffic statistics:
   Input  bytes  :                  160                  352 bps
   Output bytes  :           1159939326            646877760 bps
   Input  packets:                    2                    0 pps
   Output packets:             25216071              1757820 pps
   IPv6 transit statistics:
    Input  bytes  :                   0
    Output bytes  :                   0
    Input  packets:                   0
    Output packets:                   0
  Label-switched interface (LSI) traffic statistics:
   Input  bytes  :                    0                    0 bps
   Input  packets:                    0                    0 pps
  Dropped traffic statistics due to STP State:
   Input  bytes  :                    0
   Output bytes  :                    0
   Input  packets:                    0
   Output packets:                    0
  Input errors:
    Errors: 0, Drops: 0, Framing errors: 0, Runts: 0, Policed discards: 0, L3 
incompletes: 0, L2 channel errors: 0,
    L2 mismatch timeouts: 0, FIFO errors: 0, Resource errors: 0
  Output errors:
    Carrier transitions: 0, Errors: 0, Drops: 0, Collisions: 0, Aged packets: 0, FIFO 
errors: 0,
    HS link CRC errors: 0, MTU errors: 0, Resource errors: 0
  Egress queues: 8 supported, 4 in use
  Queue counters:       Queued packets  Transmitted packets      Dropped packets
    0 BE                      14062637             14062638                    0
    1 EF                       8437584              8437584                    0
    2 AF                       2812528              2812528                    0
    3 NC                             2                    2                    0
<…>
```

No more packet drops are happening since we increased the shaping rate to 1200M!

Use the `show interfaces queue` command to check the xe-2/1/0 interface and confirm that there are no drops reported:

```
{master}[edit]
lab@Weir-re0# run show interfaces queue xe-2/1/0
Physical interface: xe-2/1/0, Enabled, Physical link is Up
  Interface index: 145, SNMP ifIndex: 1537
```

```
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Queue: 0, Forwarding classes: BE
  Queued:
    Packets              :              42187927               976561 pps
    Bytes                :            1940644642            359374792 bps
  Transmitted:
    Packets              :              42187927                976561 pps
    Bytes                :            1940644642            359374688 bps
    Tail-dropped packets :                     0                    0 pps
    RED-dropped packets  :                     0                    0 pps
     Low                 :                     0                    0 pps
     Medium-low          :                     0                    0 pps
     Medium-high         :                     0                    0 pps
     High                :                     0                    0 pps
    RED-dropped bytes    :                     0                    0 bps
     Low                 :                     0                    0 bps
     Medium-low          :                     0                    0 bps
     Medium-high         :                     0                    0 bps
     High                :                     0                    0 bps
Queue: 1, Forwarding classes: EF
  Queued:
    Packets              :              25312760               585936 pps
    Bytes                :            1164386960            215624792 bps
  Transmitted:
    Packets              :              25312760                585937 pps
    Bytes                :            1164386960            215624896 bps
    Tail-dropped packets :                     0                    0 pps
    RED-dropped packets  :                     0                    0 pps
     Low                 :                     0                    0 pps
     Medium-low          :                     0                    0 pps
     Medium-high         :                     0                    0 pps
     High                :                     0                    0 pps
    RED-dropped bytes    :                     0                    0 bps
     Low                 :                     0                    0 bps
     Medium-low          :                     0                    0 bps
     Medium-high         :                     0                    0 bps
     High                :                     0                    0 bps
Queue: 2, Forwarding classes: AF
  Queued:
    Packets              :               8437586               195312 pps
    Bytes                :             388128956             71875000 bps
  Transmitted:
    Packets              :               8437585                195312 pps
    Bytes                :             388128910             71874896 bps
    Tail-dropped packets :                     0                    0 pps
    RED-dropped packets  :                     0                    0 pps
     Low                 :                     0                    0 pps
     Medium-low          :                     0                    0 pps
     Medium-high         :                     0                    0 pps
     High                :                     0                    0 pps
    RED-dropped bytes    :                     0                    0 bps
     Low                 :                     0                    0 bps
     Medium-low          :                     0                    0 bps
     Medium-high         :                   0                     0 bps
     High                :                   0                     0 bps
Queue: 3, Forwarding classes: NC
  Queued:
    Packets              :                   6                     0 pps
    Bytes                :                 636                   232 bps
  Transmitted:
    Packets              :                   6                      0 pps
    Bytes                :                 636                   232 bps
    Tail-dropped packets :                   0                     0 pps
    RED-dropped packets  :                   0                     0 pps
     Low                 :                   0                     0 pps
     Medium-low          :                   0                     0 pps
     Medium-high         :                   0                     0 pps
     High                :                   0                     0 pps
    RED-dropped bytes    :                   0                     0 bps
     Low                 :                   0                     0 bps
     Medium-low          :                   0                     0 bps
     Medium-high         :                   0                     0 bps
     High                :                   0                     0 bps
```

This confirms that there are no drops on the interface! So far, the drops were reported when the shaping-rate was set to 1G and 1100M, but the drops are not being reported when the shaping-rate is set to 1200M.

As another test, let's change the shaping-rate back to 1G and increase the IXIA traffic packet sizes from 64 bytes to 1500 bytes:

```
{master}[edit]
lab@Weir-re0# show class-of-service interfaces xe-2/1/0
scheduler-map CORE;
shaping-rate 1g;
unit 0 {
   classifiers {
      dscp CLASSIFIER_1;
   }
}
```

And here is the IXIA traffic packet size increase from 64 to 1500 bytes:

```
Stream #1 Best Effort
Source IP - 192.168.1.2
Destination IP - 192.168.2.2
Protocol – UDP
Frame Size – 1500 bytes
TOS – DSCP 000000 which is BE
Rate – 500Mbps

Stream #2 Assured Forwarding
Source IP - 192.168.1.2
Destination IP - 192.168.2.2
Protocol – UDP
Frame Size – 1500 bytes
```

```
TOS – DSCP 001010 which is AF
Rate – 100Mbps

Stream #3 Expedited Forwarding
Source IP - 192.168.1.2
Destination IP - 192.168.2.2
Protocol – UDP
Frame Size – 1500 bytes
TOS – DSCP 101110 which is EF
Rate – 300Mbps
```

Now let’s clear the interface statistics, and check on the ingress xe-0/3/0 statistics on Weir:

```
{master}[edit]
lab@Weir-re0# run clear interfaces statistics all

{master}[edit]
lab@Weir-re0# run show interfaces xe-0/3/0 extensive
Physical interface: xe-0/3/0, Enabled, Physical link is Up
  Interface index: 243, SNMP ifIndex: 1296, Generation: 491
  Link-level type: Ethernet, MTU: 1514, LAN-
PHY mode, Speed: 10Gbps, Loopback: None, Source filtering: Disabled,
  Flow control: Enabled
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x4000
  Link flags     : None
  CoS queues     : 8 supported, 8 maximum usable queues
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: 78:19:f7:50:20:f6, Hardware address: 78:19:f7:50:20:f6
  Last flapped   : 2012-03-03 11:22:58 PST (1d 04:02 ago)
  Statistics last cleared: 2012-03-04 15:22:21 PST (00:03:27 ago)
  Traffic statistics:
   Input  bytes  :          22985173848           889200144 bps
   Output bytes  :                    0                   0 bps
   Input  packets:             15509564               75000 pps
   Output packets:                    0                   0 pps
<…>
```

Next, check the egress xe-2/1/0 statistics and verify that no drops are being reported:

```
{master}[edit]
lab@Weir-re0# run show interfaces xe-2/1/0 extensive
Physical interface: xe-2/1/0, Enabled, Physical link is Up
  Interface index: 145, SNMP ifIndex: 1537, Generation: 393
  Link-level type: Ethernet, MTU: 1514, LAN-
PHY mode, Speed: 10Gbps, Loopback: None, Source filtering: Disabled,
  Flow control: Enabled
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x4000
  Link flags     : None
  CoS queues     : 8 supported, 8 maximum usable queues
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: 78:19:f7:50:22:e6, Hardware address: 78:19:f7:50:22:e6
  Last flapped   : 2012-03-03 11:21:04 PST (1d 04:04 ago)
  Statistics last cleared: 2012-03-04 15:22:21 PST (00:03:33 ago)
  Traffic statistics:
   Input  bytes  :                 1984                  328 bps
   Output bytes  :          23750782148            890502560 bps
   Input  packets:                   25                    0 pps
   Output packets:             16026192                75109 pps
   IPv6 transit statistics:
    Input  bytes  :                   0
    Output bytes  :                   0
    Input  packets:                   0
    Output packets:                   0
  Label-switched interface (LSI) traffic statistics:
   Input  bytes  :                    0                    0 bps
   Input  packets:                    0                    0 pps
  Dropped traffic statistics due to STP State:
   Input  bytes  :                    0
   Output bytes  :                    0
   Input  packets:                    0
   Output packets:                    0
  Input errors:
    Errors: 0, Drops: 0, Framing errors: 0, Runts: 0, Policed discards: 0, L3 
incompletes: 0, L2 channel errors: 0,
    L2 mismatch timeouts: 0, FIFO errors: 0, Resource errors: 0
  Output errors:
    Carrier transitions: 0, Errors: 0, Drops: 0, Collisions: 0, Aged packets: 0, FIFO 
errors: 0,
    HS link CRC errors: 0, MTU errors: 0, Resource errors: 0
  Egress queues: 8 supported, 4 in use
  Queue counters:       Queued packets  Transmitted packets      Dropped packets
    0 BE                     8850092              8850091                    0
    1 EF                     5310056              5310055                    0
    2 AF                     1770018              1770018                    0
    3 NC                          25                   25                    0
  <…>
```

Now let's use the `show interfaces queue` command on the egress xe-2/1/0 interface and to see if no drops were reported:

```
{master}[edit]
lab@Weir-re0# run show interfaces queue xe-2/1/0
Physical interface: xe-2/1/0, Enabled, Physical link is Up
  Interface index: 145, SNMP ifIndex: 1537
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Queue: 0, Forwarding classes: BE
  Queued:
    Packets              :              12300127                 41667 pps
    Bytes                :           18228788214             494006584 bps
  Transmitted:
    Packets              :              12300126                  41667 pps
    Bytes                :           18228786732             494006584 bps
    Tail-dropped packets :                     0                     0 pps
```

```
      RED-dropped packets  :                    0                     0 pps
       Low                 :                    0                     0 pps
       Medium-low          :                    0                     0 pps
       Medium-high         :                    0                     0 pps
       High                :                    0                     0 pps
      RED-dropped bytes    :                    0                     0 bps
       Low                 :                    0                     0 bps
       Medium-low          :                    0                     0 bps
       Medium-high         :                    0                     0 bps
       High                :                    0                     0 bps
  Queue: 1, Forwarding classes: EF
    Queued:
      Packets              :              7380077                 25000 pps
      Bytes                :          10937274114             296403288 bps
    Transmitted:
      Packets              :              7380077                  25000 pps
      Bytes                :          10937274114             296403288 bps
      Tail-dropped packets :                    0                     0 pps
      RED-dropped packets  :                    0                     0 pps
       Low                 :                    0                     0 pps
       Medium-low          :                    0                     0 pps
       Medium-high         :                    0                     0 pps
       High                :                    0                     0 pps
      RED-dropped bytes    :                    0                     0 bps
       Low                 :                    0                     0 bps
       Medium-low          :                    0                     0 bps
       Medium-high         :                    0                     0 bps
       High                :                    0                     0 bps
  Queue: 2, Forwarding classes: AF
    Queued:
      Packets              :              2460025                  8333 pps
      Bytes                :           3645757050              98800000 bps
    Transmitted:
      Packets              :              2460025                   8333 pps
      Bytes                :           3645757050              98800000 bps
      Tail-dropped packets :                    0                     0 pps
      RED-dropped packets  :                    0                     0 pps
       Low                 :                    0                     0 pps
       Medium-low          :                    0                     0 pps
       Medium-high         :                    0                     0 pps
       High                :                    0                     0 pps
      RED-dropped bytes    :                    0                     0 bps
       Low                 :                    0                     0 bps
       Medium-low          :                    0                     0 bps
       Medium-high         :                    0                     0 bps
       High                :                    0                     0 bps
  Queue: 3, Forwarding classes: NC
    Queued:
      Packets              :                   35                     0 pps
      Bytes                :                 3714                     0 bps
    Transmitted:
      Packets              :                   35                      0 pps
      Bytes                :                 3714                     0 bps
      Tail-dropped packets :                    0                     0 pps
    RED-dropped packets  :                      0                     0 pps
     Low                 :                      0                     0 pps
     Medium-low          :                      0                     0 pps
     Medium-high         :                      0                     0 pps
     High                :                      0                     0 pps
    RED-dropped bytes    :                      0                     0 bps
     Low                 :                      0                     0 bps
     Medium-low          :                      0                     0 bps
     Medium-high         :                      0                     0 bps
     High                :                      0                     0 bps
```

For the next test, let's change the packet size on the IXIA Traffic Generator from 1500 to a random size between 64 bytes and 1500 bytes. Here is the updated traffic profile now on the IXIA:

```
Stream #1 Best Effort
Source IP - 192.168.1.2
Destination IP - 192.168.2.2
Protocol – UDP
Frame Size – Random between 64 bytes and 1500 bytes
TOS – DSCP 000000 which is BE
Rate – 500Mbps

Stream #2 Assured Forwarding
Source IP - 192.168.1.2
Destination IP - 192.168.2.2
Protocol – UDP
Frame Size – Random between 64 bytes and 1500 bytes
TOS – DSCP 001010 which is AF
Rate – 100Mbps

Stream #3 Expedited Forwarding
Source IP - 192.168.1.2
Destination IP - 192.168.2.2
Protocol – UDP
Frame Size – Random between 64 bytes and 1500 bytes
TOS – DSCP 101110 which is EF
Rate – 300Mbps
```

Clear the interface statistics on Weir and then check the ingress traffic statistics:

```
{master}[edit]
lab@Weir-re0# run clear interfaces statistics all

{master}[edit]
lab@Weir-re0# run show interfaces xe-0/3/0 extensive
Physical interface: xe-0/3/0, Enabled, Physical link is Up
  Interface index: 243, SNMP ifIndex: 1296, Generation: 491
  Link-level type: Ethernet, MTU: 1514, LAN-
PHY mode, Speed: 10Gbps, Loopback: None, Source filtering: Disabled,
  Flow control: Enabled
  Device flags   : Present Running
```

```
    Interface flags: SNMP-Traps Internal: 0x4000
    Link flags     : None
    CoS queues     : 8 supported, 8 maximum usable queues
    Hold-times     : Up 0 ms, Down 0 ms
    Current address: 78:19:f7:50:20:f6, Hardware address: 78:19:f7:50:20:f6
    Last flapped   : 2012-03-03 11:22:58 PST (1d 04:10 ago)
    Statistics last cleared: 2012-03-04 15:31:21 PST (00:02:23 ago)
    Traffic statistics:
     Input  bytes  :          15742026556            879445880 bps
     Output bytes  :                    0                    0 bps
     Input  packets:             20361688               142228 pps
     Output packets:                    0                    0 pps
     <...>
```

And check the egress xe-2/1/0 interface statistics on Weir. You can see that there are still no more drops reported:

```
{master}[edit]
lab@Weir-re0# run show interfaces xe-2/1/0 extensive
Physical interface: xe-2/1/0, Enabled, Physical link is Up
  Interface index: 145, SNMP ifIndex: 1537, Generation: 393
  Link-level type: Ethernet, MTU: 1514, LAN-
PHY mode, Speed: 10Gbps, Loopback: None, Source filtering: Disabled,
  Flow control: Enabled
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x4000
  Link flags     : None
  CoS queues     : 8 supported, 8 maximum usable queues
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: 78:19:f7:50:22:e6, Hardware address: 78:19:f7:50:22:e6
  Last flapped   : 2012-03-03 11:21:04 PST (1d 04:12 ago)
  Statistics last cleared: 2012-03-04 15:31:21 PST (00:02:27 ago)
  Traffic statistics:
   Input  bytes  :                 1420                    0 bps
   Output bytes  :          16161587855            878492008 bps
   Input  packets:                   18                    0 pps
   Output packets:             20904471               142225 pps
   IPv6 transit statistics:
    Input  bytes  :                   0
    Output bytes  :                   0
    Input  packets:                   0
    Output packets:                   0
  Label-switched interface (LSI) traffic statistics:
   Input  bytes  :                    0                    0 bps
   Input  packets:                    0                    0 pps
  Dropped traffic statistics due to STP State:
   Input  bytes  :                    0
   Output bytes  :                    0
   Input  packets:                    0
   Output packets:                    0
  Input errors:
    Errors: 0, Drops: 0, Framing errors: 0, Runts: 0, Policed discards: 0, L3
incompletes: 0, L2 channel errors: 0,
    L2 mismatch timeouts: 0, FIFO errors: 0, Resource errors: 0
  Output errors:
    Carrier transitions: 0, Errors: 0, Drops: 0, Collisions: 0, Aged packets: 0, FIFO
errors: 0,
    HS link CRC errors: 0, MTU errors: 0, Resource errors: 0
  Egress queues: 8 supported, 4 in use
  Queue counters:       Queued packets  Transmitted packets      Dropped packets
    0 BE                      11662573             11662573                    0
    1 EF                       6997543              6997544                    0
    2 AF                       2332514              2332514                    0
    3 NC                            16                   16                    0
  <…>
```

Let's use the show interfaces queue command output just to be sure that there are no drops, and indeed, you'll see none are reported:

```
{master}[edit]
lab@Weir-re0# run show interfaces queue xe-2/1/0
Physical interface: xe-2/1/0, Enabled, Physical link is Up
  Interface index: 145, SNMP ifIndex: 1537
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Queue: 0, Forwarding classes: BE
  Queued:
    Packets              :              22187335                 79014 pps
    Bytes                :           17154147178             489157048 bps
  Transmitted:
    Packets              :              22187335                  79014 pps
    Bytes                :           17154147034             489156888 bps
    Tail-dropped packets :                     0                     0 pps
    RED-dropped packets  :                     0                     0 pps
     Low                 :                     0                     0 pps
     Medium-low          :                     0                     0 pps
     Medium-high         :                     0                     0 pps
     High                :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
     Low                 :                     0                     0 bps
     Medium-low          :                     0                     0 bps
     Medium-high         :                     0                     0 bps
     High                :                     0                     0 bps
Queue: 1, Forwarding classes: EF
  Queued:
    Packets              :              13312401                 47408 pps
    Bytes                :           10291129590             293180200 bps
  Transmitted:
    Packets              :              13312401                  47408 pps
    Bytes                :           10291129590             293180200 bps
    Tail-dropped packets :                     0                     0 pps
    RED-dropped packets  :                     0                     0 pps
     Low                 :                     0                     0 pps
     Medium-low          :                     0                     0 pps
     Medium-high         :                     0                     0 pps
     High                :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
     Low                 :                     0                     0 bps
```

```
      Medium-low          :                  0                    0 bps
      Medium-high         :                  0                    0 bps
      High                :                  0                    0 bps
Queue: 2, Forwarding classes: AF
  Queued:
    Packets               :            4437466                15803 pps
    Bytes                 :         3430480749             97789952 bps
  Transmitted:
    Packets               :            4437466                 15802 pps
    Bytes                 :         3430480749             97786840 bps
    Tail-dropped packets :                  0                    0 pps
    RED-dropped packets   :                  0                    0 pps
      Low                 :                  0                    0 pps
      Medium-low          :                  0                    0 pps
      Medium-high         :                  0                    0 pps
      High                :                  0                    0 pps
    RED-dropped bytes     :                  0                    0 bps
      Low                 :                  0                    0 bps
      Medium-low          :                  0                    0 bps
      Medium-high         :                  0                    0 bps
      High                :                  0                    0 bps
Queue: 3, Forwarding classes: NC
  Queued:
    Packets               :                 32                    0 pps
    Bytes                 :               3392                  232 bps
  Transmitted:
    Packets               :                 32                     0 pps
    Bytes                 :               3392                  232 bps
    Tail-dropped packets :                  0                    0 pps
    RED-dropped packets   :                  0                    0 pps
      Low                 :                  0                    0 pps
      Medium-low          :                  0                    0 pps
      Medium-high         :                  0                    0 pps
      High                :                  0                    0 pps
    RED-dropped bytes     :                  0                    0 bps
      Low                 :                  0                    0 bps
      Medium-low          :                  0                    0 bps
      Medium-high         :                  0                    0 bps
      High                :                  0                    0 bps
```

Summary

Some network engineers may notice that there are packet drops on an interface even when the traffic rate appears to be below the configured shaping rate. This is common in lab environments where someone may just configure traffic streams on a traffic generator without taking into consideration the packet headers as well as the packet size of the streams.

When configuring traffic streams on a traffic generator in the lab, it's best to create a scenario as close to the real world as possible. So the traffic streams should represent real traffic as closely as possible, for example, voice packets might be smaller than most data packets. It's also important to know how the headers of the traffic are accounted for on both the traffic generator and router side.

In this recipe, we saw packet drops when there was:

- A shaping-rate value of 1G and traffic packet size of 64 bytes.
- A shaping-rate value of 1100M and traffic packet size of 64 bytes.

The problem was *not* seen when there was:

- A shaping-rate value of 1200M and traffic packet size of 64 bytes.
- A shaping-rate value of 1G and traffic packet size of 1500 bytes.
- A shaping-rate value of 1G and traffic packet size is random.

In Junos, the class-of-service shaping-rate accounts for overhead from various encapsulations. This accounting depends on which specific hardware that the shaping-rate is being used with. It's important to understand which hardware you are using so that the correct method can be used to determine which overhead should be accounted when configuring the shaping-rate.

NOTE The following Juniper techpubs document shows which encapsulations are included in the shaping-rate calculation based on certain hardware types: http://www.juniper.net/techpubs/en_US/junos11.4/topics/usage-guidelines/cos-applying-scheduler-maps-and-shaping-rate-to-physical-interfaces-on-iq-pics.html.

NOTE Another Juniper techpubs document below talks about Trio chipset specific class-of-service features. Included, is the shaping-rate configuration. For the Trio based chipset hardware, there is an overhead of 24 bytes (20 bytes for the header and 4 bytes for the CRC): http://www.juniper.net/techpubs/en_US/junos11.4/topics/concept/trio-mpc-mic-cos-overview-cos-config-guide.html.

This overhead explains why we're seeing the packet drops on the router even though it appears the traffic rate is under the configured shaping-rate value.

First let's look at the scenario where the IXIA Traffic Generator was sending 900Mbps of total traffic. Understand that the traffic statistics in the Junos `show interfaces extensive` command accounts for only the data portion of the frame instead of the header information. IXIA was sending this 900Mbps traffic with the smallest frame size, 64 bytes. This 64 bytes includes: 14 bytes for the source and destination MAC addresses, 4 bytes of CRC, and 46 bytes of data. On Weir's ingress xe-0/3/0 interface, we saw that this port received 1757805 pps and 646872496 bps (bits per second) of traffic from the IXIA. So using these statistics to find the average data frame size of the traffic, we can use this formula:

```
Average data size in Bytes = pps / Bps
```

Now you need to convert 646872496 bps (bits per second) to Bps (Bytes per second) and to do that just divide the bps by 8, since there are 8 bits in a byte. The Bps value is: 80859062. Now the Average data size is:

```
Average data size in Bytes = 1757805 pps / 80859062 Bps
Average data size = 46 Bytes
```

You can see that the 46 Bytes value that was just calculated matches what the IXIA has configured for its data portion of the frame size!

Next, let's take a look at the real transmit rate that Ixia is sending to the router. This value will account for various overhead as well since we know that the shaping-rate takes this into consideration. Ixia is sending traffic with a frame size of 64 bytes which includes 14 bytes for the source and destination MAC addresses, 4 bytes of CRC, and 46 bytes of data. In addition to this 64 byte frame, there is a 20 byte header value that needs to be considered for the Trio chipset, so:

```
Ixia Packet size = 64 Byte frame + 20 Byte IP header
Ixia Packet size = 84 Bytes
```

So now we can use this to determine the real transmit rate that Ixia is sending. The formula will be:

```
Real Transmit Rate in bps = Ixia Packet size in bits * Router
input pps
```

We need to find the Ixia packet size in bits and we know that it is 84 Bytes, so we multiply 84 Bytes by 8 to get the bits value. This is 672. We also need the router input pps value, we know that it is 1757805 since it was used previously. So, let's solve:

```
Real Transmit Rate in bps = 672 bits * 1757805 pps = 1181244960
```

So the real transmit rate, which includes all necessary overhead for the Trio chipset is 1181244960 bps, which is about 1181 Mbps, and that's why we are seeing packet drops with a shaping-rate of 1G. From the router's point of view with the shaping-rate configuration, it believes there is more than 1G of traffic since it takes various overhead into consideration and it is dropping the packets normally.

This is why it's important to understand how the packet headers are accounted in Junos and also the traffic generator. These things need to be taken into consideration when testing and deploying class of service shaping rate configurations.