

## Chapter 2

# Class of Service Configuration Statements

This chapter shows the complete configuration statement hierarchy for class of service (CoS), listing all possible configuration statements and showing their level in the configuration hierarchy. When you are configuring the JUNOS software, your current hierarchy level is shown in the banner on the line preceding the `user@host#` prompt.

For a complete list of the JUNOS configuration statements, see the *JUNOS Hierarchy and RFC Reference*.

This chapter is organized as follows:

- [edit chassis] Hierarchy Level on page 23
- [edit class-of-service] Hierarchy Level on page 24
- [edit firewall] Hierarchy Level on page 27
- [edit interfaces] Hierarchy Level on page 28
- [edit services cos] Hierarchy Level on page 29

## [edit chassis] Hierarchy Level

---

The following CoS statements can be configured at the [edit chassis] hierarchy level. This is not a comprehensive list of statements available at the [edit chassis] hierarchy level. Only the statements that are also documented in this manual are listed here. For more information about chassis configuration, see the *JUNOS System Basics Configuration Guide*.

```
chassis {
  fpc slot-number pic pic-number {
    max-queues-per-interface (4 | 8);
    q-pic-large-buffer;
  }
}
```

**[edit class-of-service] Hierarchy Level**

```

class-of-service {
  adaptive-shapers {
    adaptive-shaper-name {
      trigger type shaping-rate (percent percentage | rate);
    }
  }
  classifiers {
    (dscp | dscp-ipv6 | exp | ieee-802.1 | inet-precedence) classifier-name {
      import (classifier-name | default);
      forwarding-class class-name {
        loss-priority level {
          code-points [ aliases ] [ 6-bit-patterns ];
        }
      }
    }
  }
  code-point-aliases {
    (dscp | dscp-ipv6 | exp | ieee-802.1 | inet-precedence) {
      alias-name bits;
    }
  }
  drop-profiles {
    profile-name {
      fill-level percentage drop-probability percentage;
      interpolate {
        drop-probability [ values ];
        fill-level [ values ];
      }
    }
  }
  fabric {
    scheduler-map {
      priority (high | low) scheduler scheduler-name;
    }
  }
  forwarding-classes {
    class class-name queue-num queue-number priority (high | low);
    queue queue-number class-name priority (high | low);
  }
  forwarding-policy {
    next-hop-map map-name {
      forwarding-class class-name {
        next-hop [ next-hop-name ];
        lsp-next-hop [ lsp-regular-expression ];
      }
    }
    class (CoS-Based Forwarding) class-name {
      classification-override {
        forwarding-class class-name;
      }
    }
  }
}

```

```

fragmentation-maps {
  map-name {
    forwarding-class class-name {
      drop-timeout milliseconds;
      fragment-threshold bytes;
      multilink-class number;
      no-fragmentation;
    }
  }
}
interfaces {
  interface-name {
    input-scheduler-map map-name;
    input-shaping-rate rate;
    scheduler-map map-name;
    scheduler-map-chassis map-name;
    shaping-rate rate;
    unit logical-unit-number {
      adaptive-shaper adaptive-shaper-name;
      classifiers {
        (dscp | dscp-ipv6 | exp | ieee-802.1 | inet-precedence)
          (classifier-name | default);
      }
      forwarding-class class-name;
      fragmentation-map map-name;
      loss-priority-maps {
        frame-relay-de (map-name | default);
      }
      input-scheduler-map map-name;
      input-shaping-rate (percent percentage | rate);
      input-traffic-control-profile profile-name shared-instance instance-name;
      output-traffic-control-profile profile-name shared-instance instance-name;
      rewrite-rules {
        dscp (rewrite-name | default);
        dscp-ipv6 (rewrite-name | default);
        exp (rewrite-name | default) protocol protocol-types;
        exp-push-push-push default;
        exp-swap-push-push default;
        frame-relay-de (rewrite-name | default);
        ieee-802.1 (rewrite-name | default) vlan-tag (outer | outer-and-inner);
        inet-precedence (rewrite-name | default);
      }
      scheduler-map map-name;
      shaping-rate rate;
      virtual-channel-group virtual-channel-group-name;
    }
  }
}
loss-priority-maps {
  frame-relay-de map-name {
    loss-priority level code-points [ values ];
  }
}
restricted-queues {
  forwarding-class class-name queue queue-number;
}

```

```

rewrite-rules {
  (dscp | dscp-ipv6 | exp | ieee-802.1 | inet-precedence) rewrite-name {
    import (rewrite-name | default);
    forwarding-class class-name {
      loss-priority level code-point (alias | bits);
    }
  }
}
routing-instances routing-instance-name {
  classifiers {
    exp (classifier-name | default);
  }
}
scheduler-maps {
  map-name {
    forwarding-class class-name scheduler scheduler-name;
  }
}
schedulers {
  scheduler-name {
    buffer-size (percent percentage | remainder | temporal microseconds);
    drop-profile-map loss-priority (any | low | medium-low | medium-high | high)
      protocol (any | non-tcp | tcp) drop-profile profile-name;
    priority priority-level;
    transmit-rate (rate | percent percentage | remainder) <exact>;
  }
}
traffic-control-profiles profile-name {
  delay-buffer-rate (percent percentage | rate);
  guaranteed-rate (percent percentage | rate);
  scheduler-map map-name;
  shaping-rate (percent percentage | rate);
}
tri-color;
virtual-channels {
  virtual-channel-name;
}
virtual-channel-groups {
  virtual-channel-group-name {
    virtual-channel-name {
      scheduler-map map-name;
      shaping-rate (percent percentage | rate);
      default;
    }
  }
}
}
}

```

## [edit firewall] Hierarchy Level

The following CoS statements can be configured at the [edit firewall] hierarchy level. This is not a comprehensive list of statements available at the [edit firewall] hierarchy level. Only the statements documented in this manual are listed here. For more information about firewall configuration, see the *JUNOS Policy Framework Configuration Guide*.

```

firewall {
  three-color-policer policer-name {
    action {
      loss-priority high then discard;
    }
    logical-interface-policer;
    single-rate {
      (color-aware | color-blind);
      committed-information-rate bps;
      committed-burst-size bytes;
      excess-burst-size bytes;
    }
    two-rate {
      (color-aware | color-blind);
      committed-information-rate bps;
      committed-burst-size bytes;
      peak-information-rate bps;
      peak-burst-size bytes;
    }
  }
}
family family-name {
  filter filter-name {
    term term-name {
      from {
        match-conditions;
      }
      then {
        dscp 0;
        forwarding-class class-name;
        loss-priority (high | low);
        virtual-channel virtual-channel-name;
        three-color-policer {
          (single-rate | two-rate) policer-name;
        }
      }
    }
  }
  simple-filter filter-name {
    term term-name {
      from {
        match-conditions;
      }
      then {
        forwarding-class class-name;
        loss-priority (high | low | medium);
      }
    }
  }
}
}

```

```

    policer policer-name {
        logical-bandwidth-policer;
        if-exceeding {
            bandwidth-limit rate;
            bandwidth-percent number;
            burst-size-limit bytes;
        }
        then {
            policer-action;
        }
    }
}

```

## [edit interfaces] Hierarchy Level

The following CoS statements can be configured at the [edit interfaces] hierarchy level. This is not a comprehensive list of statements available at the [edit interfaces] hierarchy level. Only the statements that are also documented in this manual are listed here. For more information about interface configuration, see the *JUNOS Network Interfaces Configuration Guide*.

```

interfaces {
    interface-name {
        atm-options {
            linear-red-profiles profile-name {
                high-plp-max-threshold percent;
                low-plp-max-threshold percent;
                queue-depth cells high-plp-threshold percent low-plp-threshold percent;
            }
            plp-to-clp;
            scheduler-maps map-name {
                forwarding-class class-name {
                    epd-threshold cells plp1 cells;
                    linear-red-profile profile-name;
                    priority (high | low);
                    transmit-weight (cells number | percent number);
                }
                vc-cos-mode (alternate | strict);
            }
        }
        per-unit-scheduler;
        shared-scheduler;
        schedulers number;
        unit logical-unit-number {
            atm-scheduler-map (map-name | default);
            copy-tos-to-outer-ip-header;
            family family {
                address address {
                    destination address;
                }
                filter {
                    input filter-name;
                    output filter-name;
                }
                policer {
                    input policer-name;
                }
            }
        }
    }
}

```

```

        output policer-name;
    }
    simple-filter {
        input filter-name;
    }
}
layer2-policer {
    input-policer policer-name;
    input-three-color policer-name;
    output-policer policer-name;
    output-three-color policer-name;
}
plp-to-clp;
shaping {
    (cbr rate | rtvbr peak rate sustained rate burst length |
     vbr peak rate sustained rate burst length);
}
vci vpi-identifier.vci-identifier;
}
}
}

```

## [edit services cos] Hierarchy Level

---

The following CoS statements can be configured at the [edit services cos] hierarchy level. This is not a comprehensive list of statements available at the [edit services] hierarchy level. Only the statements documented in this manual are listed here. For more information about services configuration, see the *JUNOS Services Interfaces Configuration Guide*.

```

services {
  cos {
    application-profile profile-name {
      sip-text {
        dscp (alias | bits);
        forwarding-class class-name;
      }
      sip-video {
        dscp (alias | bits);
        forwarding-class class-name;
      }
      sip-voice {
        dscp (alias | bits);
        forwarding-class class-name;
      }
    }
  }
  rule rule-name {
    match-direction (input | output | input-output);
    term term-name {
      from {
        applications [ application-names ];
        application-sets [ set-names ];
        destination-address address;
        source-address address;
      }
      then {

```

```
application-profile profile-name;  
dscp (alias | bits);  
forwarding-class class-name;  
syslog;  
(reflexive | reverse) {  
    application-profile profile-name;  
    dscp (alias | bits);  
    forwarding-class class-name;  
    syslog;  
}  
}  
}  
}  
rule-set rule-set-name {  
    [ rule rule-names ];  
}  
}  
}
```