

Chapter 18

Configuring CoS for Tunnels

For Adaptive Services, Link Services, and Tunnel Physical Interface Cards (PICs) installed on T-series and M-series platforms with enhanced Flexible PIC Concentrators (FPCs), class-of-service (CoS) information is preserved inside generic routing encapsulation (GRE) and IP-IP tunnels.

For the ES PIC installed on T-series and M-series platforms with enhanced FPCs, class-of-service information is preserved inside IPsec tunnels. For IPsec tunnels, you do not need to configure CoS, because the ES PIC copies the type-of-service (ToS) byte from the inner IP header to the GRE or IP-IP header.

To configure CoS for tunnels, you can include the following statements at the [edit class-of-service] and [edit interfaces] hierarchy level of the configuration:

```
class-of-service {
  interfaces {
    interface-name {
      unit logical-unit-number {
        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;
          ieee-802.1 (rewrite-name | default);
          inet-precedence (rewrite-name | default);
        }
      }
    }
  }
  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);
      }
    }
  }
}
interfaces {
  gre-interface-name {
    unit logical-unit-number;
    copy-tos-to-outer-ip-header;
  }
}
```

This chapter discusses the following topics:

- Configuring CoS for Tunnels on page 246
- Example: Configuring CoS for Tunnels on page 247
- Example: Configuring a GRE Tunnel to Copy TOS Bits to the Outer IP Header on page 250

Configuring CoS for Tunnels

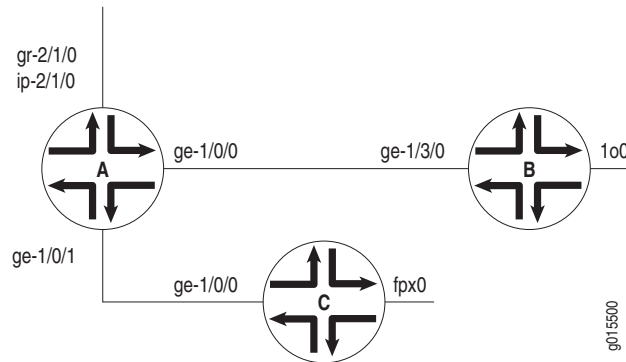
To configure CoS for GRE and IP-IP tunnels, perform the following configuration tasks:

1. To configure the tunnel, include the `tunnel` statement at the [edit interfaces *ip-fpc/pic/port* unit *logical-unit-number*] or [edit interfaces *gr-fpc/pic/port* unit *logical-unit-number*] hierarchy level.
2. To rewrite traffic on the outbound interface, include the `rewrite-rules` statement at the [edit class-of-service] and [edit class-of-service interfaces *interface-name* unit *logical-unit-number*] hierarchy levels. For GRE and IP-IP tunnels, you can configure IP precedence and DSCP rewrite rules.
3. To classify traffic on the inbound interface, you can configure a behavior aggregate (BA) classifier or firewall filter. Include the `loss-priority` and `forwarding-class` statements at the [edit firewall filter *filter-name* term *term-name* then] hierarchy level. Or you can include the `classifiers` statement at the [edit class-of-service] hierarchy level.
4. For a GRE tunnel, the default is to set the TOS bits in the outer IP header to all 0s. To copy the TOS bits from the inner IP header to the outer, include the `copy-tos-to-outer-ip-header` statement at the [edit interfaces *gr-fpc/pic/port* unit *logical-unit-number*] hierarchy level. (This inner-to-outer TOS bits copying is already the default behavior for IP-IP tunnels.)

Example: Configuring CoS for Tunnels

In Figure 14, Router A acts as a tunnel ingress device. The link between interfaces `ge-1/0/0` in Router A and `ge-1/3/0` in Router B is the GRE or IP-IP tunnel. Router A monitors the traffic received from interface `ge-1/3/0`. By way of interface `ge-1/0/0`, Router C generates traffic to Router B.

Figure 14: CoS with a Tunnel Configuration



```

Router A [edit interfaces]
ge-1/0/0 {
  unit 0 {
    family inet {
      address 10.80.0.2/24;
    }
  }
}
ge-1/0/1 {
  unit 0 {
    family inet {
      filter {
        input zf-catch-all;
      }
      address 10.90.0.2/24;
    }
  }
}
gr-2/1/0 {
  unit 0 {
    tunnel {
      source 11.11.11.11;
      destination 10.255.245.46;
    }
    family inet {
      address 21.21.21.21/24;
    }
  }
}

```

```

ip-2/1/0 {
  unit 0 {
    tunnel {
      source 12.12.12.12;
      destination 10.255.245.46;
    }
    family inet {
      address 22.22.22.22/24;
    }
  }
}

[edit routing-options]
static {
  route 1.1.1.1/32 next-hop gr-2/1/0.0;
  route 2.2.2.2/32 next-hop ip-2/1/0.0;
}

[edit class-of-service]
interfaces {
  ge-1/0/0 {
    unit 0 {
      rewrite-rules {
        inet-precedence zf-tun-rw-ipprec-00;
      }
    }
  }
}
rewrite-rules {
  inet-precedence zf-tun-rw-ipprec-00 {
    forwarding-class best-effort {
      loss-priority low code-point 000;
      loss-priority high code-point 001;
    }
    forwarding-class expedited-forwarding {
      loss-priority low code-point 010;
      loss-priority high code-point 011;
    }
    forwarding-class assured-forwarding {
      loss-priority low code-point 100;
      loss-priority high code-point 101;
    }
    forwarding-class network-control {
      loss-priority low code-point 110;
      loss-priority high code-point 111;
    }
  }
}
}

```

```

dscp zf-tun-rw-dscp-00 {
  forwarding-class best-effort {
    loss-priority low code-point 000000;
    loss-priority high code-point 001001;
  }
  forwarding-class expedited-forwarding {
    loss-priority low code-point 010010;
    loss-priority high code-point 011011;
  }
  forwarding-class assured-forwarding {
    loss-priority low code-point 100100;
    loss-priority high code-point 101101;
  }
  forwarding-class network-control {
    loss-priority low code-point 110110;
    loss-priority high code-point 111111;
  }
}

[edit firewall]
filter zf-catch-all {
  term term1 {
    then {
      loss-priority high;
      forwarding-class network-control;
    }
  }
}

```

Router B

```

[edit interfaces]
ge-1/3/0 {
  unit 0 {
    family inet {
      address 10.80.0.1/24;
    }
  }
}
lo0 {
  unit 0 {
    family inet {
      address 10.255.245.46/32;
    }
  }
}

```

```
Router C [edit interfaces]
ge-1/0/0 {
  unit 0 {
    family inet {
      address 10.90.0.1/24;
    }
  }
}

[edit routing-options]
static {
  route 1.1.1.1/32 next-hop 10.90.0.2;
  route 2.2.2.2/32 next-hop 10.90.0.2;
}
```

Example: Configuring a GRE Tunnel to Copy TOS Bits to the Outer IP Header

Unlike IP-IP tunnels, GRE tunnels do not copy the TOS bits to the outer IP header by default. To copy the inner TOS bits to the outer IP header (which is required for some tunneled routing protocols) on packets sent by the Routing Engine, include the `copy-tos-to-outer-ip-header` statement at the logical unit hierarchy level of a GRE interface. This example copies the inner TOS bits to the outer IP header on a GRE tunnel:

```
[edit interfaces]
gr-0/0/0 {
  unit 0 {
    copy-tos-to-outer-ip-header;
    family inet;
  }
}
```