

Example: Configuring Per-Unit Scheduling for Subscriber Access

In this example, a network administrator sets up a subscriber access configuration with per-unit scheduling.

1. The administrator configures the static VLAN interfaces and enables per-unit scheduling for the interfaces.

```
[edit]
interfaces {
  ge-1/1/0 {
    per-unit-scheduler;
    vlan-tagging;
    unit 100 {
      vlan-id 100;
      family inet {
        unnumbered-address lo0.0 preferred-source-address 192.100.1.1;
      }
    }
    unit 200 {
      vlan-id 200;
      family inet {
        unnumbered-address lo0.0 preferred-source-address 192.100.1.1;
      }
    }
  }
  ge-1/1/1 {
    per-unit-scheduler;
    vlan-tagging;
    unit 100 {
      vlan-id 100;
      family inet {
        unnumbered-address lo0.0 preferred-source-address 192.100.1.1;
      }
    }
    unit 200 {
      vlan-id 200;
      family inet {
        unnumbered-address lo0.0 preferred-source-address 192.100.1.1;
      }
    }
  }
  ge-1/0/1 {
    unit 0 {
      family inet {
        address 3.1.1.1/24;
      }
    }
  }
  ge-1/1/2 {
    description "wfce14 eth1 soso ge-1/1/2";
    vlan-tagging;
    gigether-options {
      no-auto-negotiation;
    }
  }
}
```

```

    }
    unit 100 {
        vlan-id 100;
        family inet {
            address 121.0.0.1/24;
        }
    }
}

```

2. The administrator configures static CoS parameters, including forwarding classes and classifiers, to be referenced in the dynamic profiles.

```

[edit]
class-of-service {
  classifiers {
    inet-precedence 8q-inet {
      forwarding-class be {
        loss-priority low code-points 000;
      }
      forwarding-class ef {
        loss-priority low code-points 001;
      }
      forwarding-class af {
        loss-priority low code-points 010;
      }
      forwarding-class nc {
        loss-priority low code-points 011;
      }
      forwarding-class voice {
        loss-priority low code-points 100;
      }
      forwarding-class video {
        loss-priority low code-points 101;
      }
      forwarding-class game {
        loss-priority low code-points 110;
      }
      forwarding-class data {
        loss-priority low code-points 111;
      }
    }
    inet-precedence 4q-inet {
      forwarding-class be {
        loss-priority low code-points [ 000 001 ];
      }
      forwarding-class ef {
        loss-priority low code-points [ 010 011 ];
      }
      forwarding-class af {
        loss-priority low code-points [ 100 101 ];
      }
      forwarding-class nc {
        loss-priority low code-points [ 110 111 ];
      }
    }
  }
}

```

```

inet-precedence 8q-drop-inet {
    forwarding-class be {
        loss-priority low code-points 000;
    }
    forwarding-class ef {
        loss-priority medium-low code-points 001;
    }
    forwarding-class af {
        loss-priority medium-high code-points 010;
    }
    forwarding-class nc {
        loss-priority high code-points 011;
    }
    forwarding-class voice {
        loss-priority low code-points 100;
    }
    forwarding-class video {
        loss-priority medium-low code-points 101;
    }
    forwarding-class game {
        loss-priority medium-high code-points 110;
    }
    forwarding-class data {
        loss-priority high code-points 111;
    }
}
inet-precedence 4q-drop-inet {
    forwarding-class be {
        loss-priority low code-points [ 000 001 ];
    }
    forwarding-class ef {
        loss-priority medium-low code-points [ 010 011 ];
    }
    forwarding-class af {
        loss-priority medium-high code-points [ 100 101 ];
    }
    forwarding-class nc {
        loss-priority high code-points [ 110 111 ];
    }
}
}
drop-profiles {
    d0 {
        fill-level 25 drop-probability 100;
        fill-level 0 drop-probability 0;
    }
    d1 {
        fill-level 50 drop-probability 100;
        fill-level 0 drop-probability 0;
    }
    d2 {
        fill-level 75 drop-probability 100;
        fill-level 0 drop-probability 0;
    }
    d3 {
        fill-level 100 drop-probability 100;
    }
}

```

```

        fill-level 0 drop-probability 0;
    }
    all {
        fill-level 0 drop-probability 0;
        fill-level 100 drop-probability 100;
    }
}
forwarding-classes {
    queue 0 be;
    queue 1 ef;
    queue 2 af;
    queue 3 nc;
    queue 4 voice;
    queue 5 video;
    queue 6 game;
    queue 7 data;
}
interfaces {
    ge-1/0/1 {
        unit 0 {
            classifiers {
                inet-precedence 8q-drop-low-high-inet;
            }
        }
    }
}
traceoptions {
    flag all;
    flag asynch;
    flag route-socket;
}
}

```

3. The administrator configures the access and service dynamic profiles to receive CoS parameters for the subscriber interfaces through RADIUS.

```

[edit]
dynamic-profiles {
    subscriber {
        interfaces {
            "$junos-interface-ifd-name" {
                unit "$junos-underlying-interface-unit" {
                    family inet;
                }
            }
        }
    }
    class-of-service {
        traffic-control-profiles {
            zero {
                scheduler-map "$junos-cos-scheduler-map";
                shaping-rate "$junos-cos-shaping-rate";
                guaranteed-rate "$junos-cos-guaranteed-rate";
                delay-buffer-rate "$junos-cos-delay-buffer-rate";
            }
        }
    }
    interfaces {

```



```

variables {
    fc_1 default-value be;
    sch_1 default-value be_sch;
    sch-tx_1 default-value 20000000;
    sch-bs_1 default-value 10;
    sch-pri_1 default-value high;
    sch-drop-low_1 default-value d3;
    sch-drop-med-low_1 default-value d2;
    sch-drop-med-high_1 default-value d1;
    sch-drop-high_1 default-value d0;
    sch-drop-any_1 default-value d3;
    fc_2 default-value af;
    sch_2 default-value af_sch;
    sch-tx_2 default-value 10;
    sch-bs_2 default-value 10;
    sch-pri_2 default-value high;
    sch-drop-low_2 default-value d3;
    sch-drop-med-low_2 default-value d2;
    sch-drop-med-high_2 default-value d1;
    sch-drop-high_2 default-value d0;
    sch-drop-any_2 default-value d3;
    fc_3 default-value voice;
    sch_3 default-value voice_sch;
    sch-tx_3 default-value 20000000;
    sch-bs_3 default-value 10;
    sch-pri_3 default-value high;
    sch-drop-low_3 default-value d3;
    sch-drop-med-low_3 default-value d2;
    sch-drop-med-high_3 default-value d1;
    sch-drop-high_3 default-value d0;
    sch-drop-any_3 default-value d3;
    fc_4 default-value game;
    sch_4 default-value game_sch;
    sch-tx_4 default-value 10;
    sch-bs_4 default-value 10;
    sch-pri_4 default-value high;
    sch-drop-low_4 default-value d3;
    sch-drop-med-low_4 default-value d2;
    sch-drop-med-high_4 default-value d1;
    sch-drop-high_4 default-value d0;
    sch-drop-any_4 default-value d3;
    scheduler-map default-value all_smap;
}
class-of-service {
    scheduler-maps {
        "$scheduler-map" {
            forwarding-class "$fc_1" scheduler "$sch_1";
            forwarding-class "$fc_2" scheduler "$sch_2";
            forwarding-class "$fc_3" scheduler "$sch_3";
            forwarding-class "$fc_4" scheduler "$sch_4";
        }
    }
    schedulers {
        "$sch_1" {
            transmit-rate "$sch-tx_1";
            buffer-size percent "$sch-bs_1";

```

```

    priority "$sch-pri_1";
    drop-profile-map loss-priority low protocol any drop-profile
        "$sch-drop-low_1";
    drop-profile-map loss-priority medium-low protocol any drop-profile
        "$sch-drop-med-low_1";
    drop-profile-map loss-priority medium-high protocol any drop-profile
        "$sch-drop-med-high_1";
    drop-profile-map loss-priority high protocol any drop-profile
        "$sch-drop-high_1";
}
"$sch_2" {
    transmit-rate percent "$sch-tx_2";
    buffer-size percent "$sch-bs_2";
    priority "$sch-pri_2";
    drop-profile-map loss-priority low protocol any drop-profile
        "$sch-drop-low_2";
    drop-profile-map loss-priority medium-low protocol any drop-profile
        "$sch-drop-med-low_2";
    drop-profile-map loss-priority medium-high protocol any drop-profile
        "$sch-drop-med-high_2";
    drop-profile-map loss-priority high protocol any drop-profile
        "$sch-drop-high_2";
}
"$sch_3" {
    transmit-rate "$sch-tx_3";
    buffer-size percent "$sch-bs_3";
    priority "$sch-pri_3";
    drop-profile-map loss-priority low protocol any drop-profile
        "$sch-drop-low_3";
    drop-profile-map loss-priority medium-low protocol any drop-profile
        "$sch-drop-med-low_3";
    drop-profile-map loss-priority medium-high protocol any drop-profile
        "$sch-drop-med-high_3";
    drop-profile-map loss-priority high protocol any drop-profile
        "$sch-drop-high_3";
}
"$sch_4" {
    transmit-rate percent "$sch-tx_4";
    buffer-size percent "$sch-bs_4";
    priority "$sch-pri_4";
    drop-profile-map loss-priority low protocol any drop-profile
        "$sch-drop-low_4";
    drop-profile-map loss-priority medium-low protocol any drop-profile
        "$sch-drop-med-low_4";
    drop-profile-map loss-priority medium-high protocol any drop-profile
        "$sch-drop-med-high_4";
    drop-profile-map loss-priority high protocol any drop-profile
        "$sch-drop-high_4";
}
}
}
}
service_2 {
    variables {
        fc_1 default-value be;
        sch_1 default-value be_sch;
    }
}

```

```

sch-tx_1 default-value 10;
sch-bs_1 default-value 10;
sch-pri_1 default-value high;
sch-drop-low_1 default-value d3;
sch-drop-med-low_1 default-value d2;
sch-drop-med-high_1 default-value d1;
sch-drop-high_1 default-value d0;
sch-drop-any_1 default-value d3;
scheduler-map default-value all_smap;
}
class-of-service {
  scheduler-maps {
    "$scheduler-map" {
      forwarding-class "$fc_1" scheduler "$sch_1";
    }
  }
  schedulers {
    "$sch_1" {
      transmit-rate percent "$sch-tx_1";
      buffer-size percent "$sch-bs_1";
      priority "$sch-pri_1";
      drop-profile-map loss-priority low protocol any drop-profile
        "$sch-drop-low_1";
      drop-profile-map loss-priority medium-low protocol any drop-profile
        "$sch-drop-med-low_1";
      drop-profile-map loss-priority medium-high protocol any drop-profile
        "$sch-drop-med-high_1";
      drop-profile-map loss-priority high protocol any drop-profile
        "$sch-drop-high_1";
    }
  }
}
}
}

```

4. The network administrator configures DHCP and RADIUS to grant access and services to the interfaces referenced by the `subscriber` dynamic profile.

```

[edit]
forwarding-options {
  dhcp-relay {
    traceoptions {
      file size 1g;
      flag all;
    }
  }
  dynamic-profile subscriber aggregate-clients replace;
  server-group {
    subscriber-server {
      3.1.1.2;
    }
  }
  active-server-group subscriber-server;
  group relay-0 {
    authentication {
      password pwd0;
      username-include {

```



```

        user-prefix user0;
        mac-address;
    }
}
interface ge-1/1/0.100;
interface ge-1/1/0.200;
}
}
radius-server {
    121.0.0.11 secret "$9$mPF/u0lcrv1RvL7V4oik.Pz3/Ct0IE"; ## SECRET-DATA
}
profile subscriber-profile {
    authentication-order radius;
    radius {
        authentication-server 121.0.0.11;
        accounting-server 121.0.0.11;
    }
    radius-server {
        121.0.0.11 secret "$9$.mz6pu1hyKBIK8xdg4jHqmQF69A01R"; ##
        SECRET-DATA
    }
    accounting {
        order radius;
        statistics time;
    }
}
}

```

Related Topics ■ [Configuring Per-Unit Scheduling in a Dynamic Profile for Subscriber Access](#)

Published: 2010-04-15