Appendix A. Configuration examples

A1. CISCO IP-SLA Monitoring Probe Configuration Example

Cisco® IP-SLA Probe Command Line Configuration used on JANET

Monitoring probes at sites are configured with the Cisco® IOS command:

!

tr responder

!

This enables the IP-SLA process, allowing the router to receive, process, and return probes back to the CMP. For IOS version 12.4 this command became ‘ip sla monitor responder’ and the related commands changed also.

The CMP is configured to send the probes. Each probe consists of a set of command lines, one set to initialize the probe settings and then an ‘rtr schedule’ command to start the probe running.

!

! First create an IP-SLA/RTR probe, with an ID of 150

!

rtr 150

!

! Set the type and arguments for the probe.

! In this case it is a jitter probe which sends 50 packets, 20ms apart (default)

! All IP addresses have been changed in this example

!

type jitter dest-ipaddr 10.10.10.10 dest-port 22200 num-packets 50

```
!
! The owner command is simply a text field - so is used as a description
!
owner Swansea to Lancaster BE
! The tag field is used to ties the probe to a database table where the data is ! stored
tag qos_swan2lanc_be_saa
!
! Now we need to start the probe, and let it run forever
!
rtr schedule 150 life forever start-time now
!
! The same is repeated now for LBE and Premium marked probes
! Note the additional tos command in each
!
rtr 151
type jitter dest-ipaddr 10.10.10.10 dest-port 22201 num-packets 50
tos 0x20
owner Swansea to Lancaster LBE
tag qos_swan2lanc_lbe_saa
rtr schedule 151 life forever start-time now  !
rtr 155
type jitter dest-ipaddr 10.10.10.10 dest-port 22205 num-packets 50
tos 0xB8
owner Swansea to Lancaster EF
tag qos_swan2lanc_ef_saa
rtr schedule 155 life forever start-time now
```

```
!
```

A2. Juniper M7i QoS Configuration Example

Step-by-step configuration for QoS used at Lancaster on the CLEO Network

In order to move data into various queues the Juniper requires a classifier:

```
class-of-service {
classifiers {
  dscp cp-classifier {
   import default;
forwarding-class network-control {
    loss-priority low code-points 100010;
   }
   forwarding-class premium {
    loss-priority low code-points 011010;
   }
   forwarding-class best-effort {
    loss-priority low code-points [ 010010 000000 ];
   }
   forwarding-class less-than-best-effort {
    loss-priority high code-points 001010;
   }
  }
}
}
```

These classes must then be allocated to a queue:

```
class-of-service {
forwarding-classes {
```

```
   queue 0 premium;
queue 1 best-effort;
queue 2 less-than-best-effort;
queue 3 network-control;
}
}
```

Schedulers are also required in order to provide prioritisation:

```
class-of-service {
schedulers {
   nc-scheduler {
    transmit-rate percent 15;
buffer-size percent 15;
priority high;
   }
   p-scheduler {
    transmit-rate percent 20;
buffer-size percent 20;
priority high;
   }
   be-scheduler {
    transmit-rate percent 50;
buffer-size percent 50;
    priority high;
   }
   lbe-scheduler {
    transmit-rate percent 15;
```

```
buffer-size percent 15;
priority low;
  }
}
}
```

These schedulers are then mapped to the forwarding classes:

```
class-of-service {
scheduler-maps {
  cos-test {
   forwarding-class network-control scheduler nc-scheduler;
forwarding-class premium scheduler p-scheduler;
forwarding-class best-effort scheduler be-scheduler;
forwarding-class less-than-best-effort scheduler lbescheduler;
  }
}
}
```

Finally each interface must be linked to a scheduler map and the classifier:

```
class-of-service {
interfaces {
  ge-0/1/0 {
   scheduler-map cos-test;
unit 0 {
    classifiers {
     dscp cp-classifier;
    }
   }
```

```
    }
    fe-1/3/0 {
     scheduler-map cos-test;
  unit 0 {
      classifiers {
       dscp cp-classifier;
      }
     }
    }
  }
  }
```

A3. Cisco® 7401 QoS Configuration Example

Sample configuration for Cisco® 7401 BAR in C&NLMAN

```
! lu-bar QoS config
! There seems to be disagreement on whether you need to enable cef
! for QoS
! ip cef
!
! ports are as follows,
! GigabitEthernet0/0 - Uplink to JANET
! GigabitEthernet0/1 - Downlink to Campus Network
!
! Classify the traffic passing through the router
! AF - Assured forwarding, router updates traffic.
! IN - Inbound traffic from CANLMAN
```

```
! PREMIUM - Traffic allowed to mask DSCP values to 46  ! REMARK - Catch anything else and remark its DSCP value
!
class-map match-all AF
  match ip dscp 48
match access-group 199 in
class-map match-all IN
match ip dscp 46
match input-interface GigabitEthernet0/0
class-map match-any PREMIUM
match ip dscp 46
match access-group 190 in
class-map match-any REMARK
match input-interface GigabitEthernet0/1
!
! Create policies for each of the classes of traffic defined above.
! PREMIUM - gets 50Mbit/s (5% of 1gig)
! AF - gets 15% of remaining
! REMARK - sets the DSCP of any un-authorised traffic to 0
! DEFAULT - catches anything else and remarks to 0
!
policy-map LANCS
  class  PREMIUM
    priority 50000
  class  AF
    bandwidth percent 20
  class  REMAP
```

```
    set ip dscp 0
  class  DEFAULT
    set ip dscp 0
policy-map CANLMAN
  class  IN
    priority 50000
!
! Then the policies need to be applied to the interfaces
! WARNING, don't just copy & paste this bit as it will break other
! interface settings
!
interface GigabitEthernet0/0
  service-policy output LANCS
interface GigabitEthernet0/1
service-policy output CANLMAN
```

A4. Catalyst 3550 QoS Configuration Example

Typical Cisco® 3550 QoS Configuration used by a school in SSDN

```
ip multicastrouting
mls qos map cosdscp 0 8 16 26 32 46 48 56
mls qos minreserve 5 170
mls qos minreserve 6 10
mls qos minreserve 7 65
mls qos minreserve 8 26
mls qos
!
!
```

interface FastEthernet0/1

description Link to Hub Secondary

no switchport

ip address 10.250.250.50 255.255.255.252

ip pim densemode

loadinterval 30

speed 10

mls qos trust cos

auto qos voip trust

wrrqueue bandwidth 20 1 80 1

wrrqueue minreserve 1 5

wrrqueue minreserve 2 6

wrrqueue minreserve 3 7

wrrqueue minreserve 4 8

wrrqueue cosmap 1 0 1 2 4

wrrqueue cosmap 3 3 6 7

wrrqueue cosmap 4 5

priorityqueue out

!

!interface FastEthernet0/22

description Voice Vlan

switchport trunk encapsulation dot1q

switchport trunk native vlan 10

switchport mode trunk

switchport voice vlan 100

no ip address

shutdown

mls qos trust device ciscophone

mls qos trust cos

auto qos voip ciscophone

wrrqueue bandwidth 20 1 80 1

wrrqueue minreserve 1 5

wrrqueue minreserve 2 6

wrrqueue minreserve 3 7

wrrqueue minreserve 4 8

wrrqueue cosmap 1 0 1 2 4

wrrqueue cosmap 3 3 6 7

wrrqueue cosmap 4 5

priorityqueue out

spanningtree portfast

!

interface FastEthernet0/23

description Voice Vlan

switchport trunk encapsulation dot1q

switchport trunk native vlan 10

switchport mode trunk

switchport voice vlan 100

no ip address

mls qos trust device ciscophone

mls qos trust cos

auto qos voip ciscophone

wrrqueue bandwidth 20 1 80 1

wrrqueue minreserve 1 5

wrrqueue minreserve 2 6

wrrqueue minreserve 3 7

wrrqueue minreserve 4 8

wrrqueue cosmap 1 0 1 2 4

wrrqueue cosmap 3 3 6 7

wrrqueue cosmap 4 5

priorityqueue out

spanningtree portfast

!

interface FastEthernet0/24

description Voice Vlan

switchport trunk encapsulation dot1q

switchport trunk native vlan 10 switchport mode trunk

switchport voice vlan 100

no ip address

mls qos trust device ciscophone

mls qos trust cos

auto qos voip ciscophone

wrrqueue bandwidth 20 1 80 1

wrrqueue minreserve 1 5

wrrqueue minreserve 2 6

wrrqueue minreserve 3 7

wrrqueue minreserve 4 8

wrrqueue cosmap 1 0 1 2 4

wrrqueue cosmap 3 3 6 7

wrrqueue cosmap 4 5

priorityqueue out

spanningtree portfast

!
