

ASR 920에서 마이크로버스트를 처리하는 방법, 바이트와 백분율 간의 차이(Queue-Limit 사용)

목차

[소개](#)

[사전 요구 사항](#)

[요구 사항](#)

[사용되는 구성 요소](#)

[배경 정보](#)

[큐 제한 구성](#)

[RSP3 및 NCS 520과의 차이점](#)

[QL\(Queue-Limit\) 랩 테스트, 바이트와 백분율 간의 차이](#)

[테스트 범위 및 매개변수](#)

[테스트 절차](#)

[예](#)

[랩 테스트 결과](#)

[QL 퍼센트 대 64바이트 패킷 바이트](#)

[QL 퍼센트 대 200바이트 패킷 바이트](#)

[QL 퍼센트 대 300바이트 패킷 바이트](#)

[QL 퍼센트 대 518바이트 패킷 바이트](#)

[QL 퍼센트 대 800바이트 패킷 바이트](#)

[QL 퍼센트 대 1024바이트 패킷 바이트](#)

[QL 퍼센트 대 1400바이트 패킷 바이트](#)

[드롭과 패킷 크기 간의 관계](#)

[동등한 바이트와 백분율 값의 차이](#)

[결론](#)

[실제 트래픽 시나리오에서 백분율 값을 근사화하는 방법](#)

[마이크로버스트 버퍼 사용량을 확인하는 방법](#)

소개

이 문서에서는 인터페이스 출력 패킷 삭제의 원인이 되는 ASR 920 라우터의 마이크로버스트를 처리하는 방법에 대해 설명합니다. 자세히 보면 queue-limit 명령을 사용하여 바이트 사용량과 백분율 간의 차이를 보여 줍니다.

사전 요구 사항

요구 사항

다음 주제에 대한 지식을 보유하고 있으면 유용합니다.

ASR 920 series 라우터

사용되는 구성 요소

이 문서의 정보는 소프트웨어 버전 Cisco IOS XE 16.9.6을 실행하는 ASR 920 라우터를 기반으로 합니다.

IXIA는 랩 테스트의 트래픽 생성기로 사용됩니다.

이 문서의 정보는 특정 랩 환경의 디바이스를 토대로 작성되었습니다. 이 문서에 사용된 모든 디바이스는 초기화된(기본) 컨피그레이션으로 시작되었습니다. 현재 네트워크가 작동 중인 경우 모든 명령의 잠재적인 영향을 미리 숙지하시기 바랍니다.

배경 정보

마이크로버스트는 네트워크 트래픽의 소규모 급증이라고 합니다. 예를 들어 통신 사업자 액세스 네트워크에서는 트래픽 흐름이 10GE(기가비트 이더넷)와 같은 고속 인터페이스에서 라우터로 들어 오고 1GE와 같은 저속 인터페이스를 통해 나가는 속도 불일치 시나리오에서 일반적으로 발생합니다.

ASR 920 라우터에서 마이크로버스트가 일으키는 가장 일반적인 문제는 출력 방향의 인터페이스 패킷 손실입니다. 이는 매우 짧은 시간(밀리초 단위) 동안 수신 인터페이스에서 발신 인터페이스의 속도보다 높은 버스트가 있을 때 발생합니다. 이 시간 동안 패킷을 버퍼링해야 합니다.

ASR 920/RSP2 플랫폼에서 각 1GE 인터페이스의 큐에 할당된 기본 버퍼는 48KB이며, 각 10GE 인터페이스의 큐에 할당된 기본 버퍼는 120KB입니다.

여기에 11.75MB의 공유 버퍼가 있습니다. 기본 인터페이스 버퍼가 패킷 버스트를 수용하기에 충분하지 않을 경우 공유 버퍼를 사용할 수 있습니다.

인터페이스에 공유 버퍼를 사용하려면 해당 인터페이스에서 queue-limit 매개변수를 정의하는 QoS 정책을 구성해야 합니다.

큐 제한 구성

다음은 인터페이스 GE 0/0/1의 queue-limit 컨피그레이션 예입니다.

```
class-map match-all DUMMY
match qos-group 99 <--- it can be any unused group
```

```
policy-map QUEUE-LIMIT
class DUMMY
class class-default
queue-limit percent 5
```

```
interface GigabitEthernet 0/0/1
service-policy output QUEUE-LIMIT
```

이 정책은 어떤 트래픽과도 일치하지 않으므로, 이 정책에도 영향을 미치지 않습니다. 이 정책은 인터페이스 GE 0/0/1의 대기열 버퍼를 늘리는 것입니다.

비 leaf 클래스에서 queue-limit을 구성하는 것은 지원되지 않으므로 **match qos-group 99** 명령이 필요합니다. 따라서 leaf class-default 아래에서 queue-limit 명령을 구성하려면 가상의 class-map을 상위 클래스로 사용해야 합니다.

```
ASR-920-1(config)#policy-map QUEUE-LIMIT
ASR-920-1(config-pmap)#class class-default
ASR-920-1(config-pmap-c)# queue-limit percent 5
QoS: queue-limit command not supported in non-leaf classes
queue-limit: platform params check fail
```

queue-limit은 다음과 같은 다양한 방법으로 설정할 수 있습니다.

```
(config-pmap-c)#queue-limit ?
<1-2097152> in bytes, <1-1677721> in us, <1-8192000> in packets by default
percent % of threshold
```

매개변수와 값은 네트워크 요구 사항에 따라 선택해야 합니다.

이러한 매개변수와 값 변경이 버퍼 사용량에 미치는 영향을 파악하기 위해 다음 섹션에 제시된 Lab 테스트를 참조할 수 있습니다.

참고: 포트 채널 인터페이스 아래에서는 QoS 정책을 구성할 수 없으며, 포트 채널의 물리적 인터페이스 부분에서만 구성할 수 있습니다

```
ASR-920-1(config)#interface port-channel 2
ASR-920-1(config-if)#service-policy output qos-tac
QoS: Configuration failed. Policy-map with Queueing actions not supported on EC main-
interface/EFP
QoS: Configuration errors for policymap qos-tac
```

RSP3 및 NCS 520과의 차이점

[Cisco RSP3 모듈 QoS 기능:](#)

- RSP3 모듈에는 NPU당 4GB 외부 패킷 버퍼가 있음
- RSP3 모듈에서 48000 큐 지원
- 기본적으로 RSP3 모듈은 큐당 최대 1MB의 큐 제한을 지원합니다
- 큐 제한 백분율은 총 버퍼 중 1GB에서 제외된 것으로 간주됩니다.

RSP3 수퍼바이저 및 NCS 520이 있는 라우터의 경우 구성할 수 있는 qos 그룹의 수는 0-7로 제한됩니다.

```
ASR-903-1(config)#class-map match-all qos-tac
ASR-903-1(config-cmap)#match qos-group ?
<0-7> Qos Group value
```

NCS 520에는 기본적으로 모든 인터페이스 간에 2MB의 대기열 버퍼가 공유되어 있으며, queue-limit을 사용하는 정책 맵이 구성되어 있으면 외부 2GB 버퍼에 액세스할 수 있습니다. queue-limit에 대한 바이트 및 us 매개변수도 차이가 있습니다.

```
ASR-520-1(config-pmap-c)# queue-limit ?
<1-8192000> in bytes, <1-40000> in us, <1-8192000> in packets by default
percent % of threshold
```

QL(Queue-Limit) 랩 테스트, 바이트와 백분율 간의 차이

테스트 범위 및 매개변수

앞에서 관찰한 것처럼 queue-limit(바이트)에 대해 구성 가능한 최대값은 2097152이며, 이는 ASR 920 플랫폼의 공유 버퍼(11.75MB ~ 45898 * 256바이트)의 약 18%입니다. 그러나 queue-limit을 퍼센트 단위로 구성할 경우 최대 100%까지 진행할 수 있습니다. 따라서 백분율과 바이트를 동등한 값과 비교하기 위해 테스트에서는 117498~2097152바이트의 바이트 값과 1%에서 18%의 queue-limit 백분율 값을 사용합니다.

```
queue-limit percent 1 <=> queue-limit 117498 bytes
queue-limit percent 2 <=> queue-limit 234996 bytes
queue-limit percent 3 <=> queue-limit 352494 bytes
queue-limit percent 4 <=> queue-limit 469992 bytes
queue-limit percent 5 <=> queue-limit 587490 bytes
queue-limit percent 6 <=> queue-limit 704988 bytes
queue-limit percent 7 <=> queue-limit 822486 bytes
queue-limit percent 8 <=> queue-limit 939984 bytes
queue-limit percent 9 <=> queue-limit 1057482 bytes
queue-limit percent 10 <=> queue-limit 1174980 bytes
queue-limit percent 11 <=> queue-limit 1292478 bytes
queue-limit percent 12 <=> queue-limit 1409976 bytes
queue-limit percent 13 <=> queue-limit 1527474 bytes
queue-limit percent 14 <=> queue-limit 1644972 bytes
queue-limit percent 15 <=> queue-limit 1762470 bytes
queue-limit percent 16 <=> queue-limit 1879968 bytes
queue-limit percent 17 <=> queue-limit 1997466 bytes
queue-limit percent 18 <=> queue-limit 2097152 bytes
```

36개의 정책 맵이 구성됩니다. 18은 1%~18% 범위의 queue-limit 값을, 나머지 18은 117498~2097152바이트 범위의 queue-limit 값을 가집니다.

```
policy-map QUEUE-LIMIT-PERCENT-X
class DUMMY
class class-default
  queue-limit percent X
```

```
policy-map QUEUE-LIMIT-BYTES-X
class DUMMY
class class-default
  queue-limit Y bytes
```

=> X values range from 1 to 18
=> Y values range from 117498 to 2097152

각 정책은 IXIA에서 생성되는 동일한 마이크로버스트 트래픽에 대해 테스트됩니다. 이 트래픽은 ASR 920의 10GE 포트에 도착하고 동일한 라우터의 1GE 포트에서 나갑니다.

버스트는 5초 간격으로 4Gbps의 20000 패킷으로 구성됩니다. 특정 패킷 크기에 지정된 버스트 기간입니다.

```
1280000 bytes at 64 packet size, Burst duration: 0.00256 second
4000000 bytes at 200 packet size, Burst duration: 0.008 second
6000000 bytes at 300 packet size, Burst duration: 0.012 second
10360000 bytes at 518 packet size, Burst duration: 0.02072 second
16000000 bytes at 800 packet size, Burst duration: 0.032 second
20480000 bytes at 1024 packet size, Burst duration: 0.04096 second
28000000 bytes at 1400 packet size, Burst duration: 0.056 second
```

테스트 절차

1단계. 아웃바운드 인터페이스 아래에서 정책 맵 QUEUE-LIMIT-BYTES-X(X=1)를 바이트

Y=117498과 함께 적용합니다.

2단계. 마이크로버스트 트래픽을 1분 동안 실행합니다.

3단계. 출력에 있는 총 패킷 수와 삭제된 패킷 수를 측정합니다.

4단계. 삭제된 패킷과 총 출력 패킷 간의 비율을 계산합니다.

5단계. 1단계부터 한 번 반복합니다. 이번에는 policy-map QUEUE-LIMIT-PERCENT-X를 queue-limit percent X와 함께 사용합니다. 여기서 X=1입니다.

6단계. 정책 맵 이름 및 백분율 값, Y=Y+117498바이트에 대해 1단계에서 X=X+1로 반복합니다. X=18, Y=2097152까지 반복합니다.

예

QUEUE-LIMIT-BYTES-1 정책 맵을 사용한 측정:

```
ASR-920-1#show int Gi0/0/1
```

```
GigabitEthernet0/0/1 is up, line protocol is up
  Hardware is 24xGE-4x10GE-FIXED-S, address is 70df.2f2f.ed01 (bia 70df.2f2f.ed01)
  Internet address is 10.12.10.47/31
  MTU 8900 bytes, BW 1000000 Kbit/sec, DLY 10 usec,
    reliability 255/255, txload 4/255, rxload 4/255
  Encapsulation ARPA, loopback not set
  Keepalive set (10 sec)
  Full Duplex, 1000Mbps, link type is auto, media type is T
  output flow-control is unsupported, input flow-control is on
  Carrier delay is 0 msec
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input 00:00:01, output 00:00:02, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/375/0/0 (size/max/drops/flushes); Total output drops: 684263427
  Queueing strategy: Class-based queueing
  Output queue: 0/40 (size/max)
  30 second input rate 19475000 bits/sec, 19533 packets/sec
  30 second output rate 19157000 bits/sec, 13356 packets/sec
  5064106237 packets input, 4333296255278 bytes, 0 no buffer
  Received 29 broadcasts (0 IP multicasts)
  0 runs, 0 giants, 0 throttles
  0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
  0 watchdog, 726180 multicast, 0 pause input
  7829367523 packets output, 4217074973677 bytes, 0 underruns
  0 output errors, 0 collisions, 3 interface resets
  0 unknown protocol drops
  0 babbles, 0 late collision, 0 deferred
  0 lost carrier, 0 no carrier, 0 pause output
  0 output buffer failures, 0 output buffers swapped out
```

```
ASR-920-1#show policy-map int Gi0/0/1 output
```

```
GigabitEthernet0/0/1
```

```
Service-policy output: QUEUE-LIMIT-BYTES-1
```

```
Class-map: DUMMY (match-all)
```

```
  0 packets, 0 bytes
```

```
  30 second offered rate 0000 bps
```

Match: qos-group 99

Class-map: class-default (match-any)
1044078 packets, 73085460 bytes
30 second offered rate 9759000 bps, drop rate 0000 bps
Match: any

queue limit 117498 bytes
(queue depth/total drops/no-buffer drops) 0/0/0
(pkts output/bytes output) 0/0

마이크로버스트 트래픽 1분 후 측정:

ASR-920-1#show int Gi0/0/1

GigabitEthernet0/0/1 is up, line protocol is up
Hardware is 24xGE-4x10GE-FIXED-S, address is 70df.2f2f.ed01 (bia 70df.2f2f.ed01)
Internet address is 10.12.10.47/31
MTU 8900 bytes, BW 1000000 Kbit/sec, DLY 10 usec,
reliability 255/255, txload 2/255, rxload 3/255
Encapsulation ARPA, loopback not set
Keepalive set (10 sec)
Full Duplex, 1000Mbps, link type is auto, media type is T
output flow-control is unsupported, input flow-control is on
Carrier delay is 0 msec
ARP type: ARPA, ARP Timeout 04:00:00
Last input 00:00:01, output 00:00:01, output hang never
Last clearing of "show interface" counters never
Input queue: 0/375/0/0 (size/max/drops/flushes); Total output drops: 684561562
Queueing strategy: Class-based queueing
Output queue: 0/40 (size/max)
30 second input rate 13981000 bits/sec, 19643 packets/sec
30 second output rate 11256000 bits/sec, 12784 packets/sec
5064715137 packets input, 4333338878716 bytes, 0 no buffer
Received 29 broadcasts (0 IP multicasts)
0 runts, 0 giants, 0 throttles
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
0 watchdog, 726190 multicast, 0 pause input
7829753878 packets output, 4217102018968 bytes, 0 underruns
0 output errors, 0 collisions, 3 interface resets
0 unknown protocol drops
0 babbles, 0 late collision, 0 deferred
0 lost carrier, 0 no carrier, 0 pause output
0 output buffer failures, 0 output buffers swapped out

ASR-920-1#show policy-map int Gi0/0/1 output

GigabitEthernet0/0/1

Service-policy output: QUEUE-LIMIT-BYTES-1

Class-map: DUMMY (match-all)
0 packets, 0 bytes
30 second offered rate 0000 bps
Match: qos-group 99

Class-map: class-default (match-any)
1847215 packets, 129305050 bytes
30 second offered rate 10804000 bps, drop rate 0000 bps
Match: any

```
queue limit 117498 bytes
(queue depth/total drops/no-buffer drops) 0/387570/0
(pkts output/bytes output) 656508/45955560
```

패킷이 델타를 삭제합니다. $684561562 - 684263427 = 298135$
총 패킷 출력 델타: $7829753878 - 7829367523 = 386355$
패킷 삭제 및 패킷 아웃 비율: $298135/386355 = 77\%$

랩 테스트 결과

설명한 대로 36개의 정책 맵이 테스트됩니다. 1%~18% 범위의 queue-limit 값으로 구성된 18은 117498~2097152바이트 범위의 queue-limit 값으로 구성된 다른 18개의 정책과 비교하여 테스트됩니다. 각 정책 맵은 IXIA에서 생성된 동일한 마이크로버스트 트래픽에 대해 테스트됩니다.

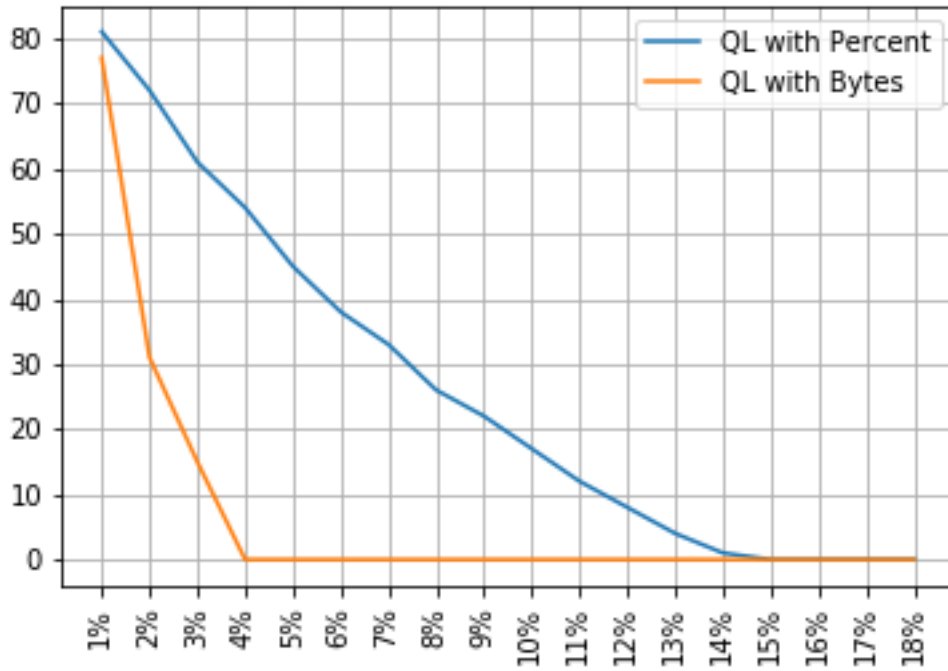
이 섹션에서는 패킷 크기가 다른 결과를 바이트 단위로 확인하기 위해 7회 반복해서 이 테스트의 결과를 표시합니다. 64, 200, 300, 518, 800, 1024, 1400.

읽기가 용이하도록 각 패킷 크기에 대해 결과가 테이블에 노출된 다음 그래프로 표시됩니다.

QL 퍼센트 대 64바이트 패킷 바이트

Policy-map		Drop rate with PC	Drop rate with Bytes
QL Bytes/Percent	1	81%	77%
QL Bytes/Percent	2	72%	31%
QL Bytes/Percent	3	61%	15%
QL Bytes/Percent	4	54%	0%
QL Bytes/Percent	5	45%	0%
QL Bytes/Percent	6	38%	0%
QL Bytes/Percent	7	33%	0%
QL Bytes/Percent	8	26%	0%
QL Bytes/Percent	9	22%	0%
QL Bytes/Percent	10	17%	0%
QL Bytes/Percent	11	12%	0%
QL Bytes/Percent	12	8%	0%
QL Bytes/Percent	13	4%	0%
QL Bytes/Percent	14	1%	0%
QL Bytes/Percent	15	0%	0%
QL Bytes/Percent	16	0%	0%
QL Bytes/Percent	17	0%	0%
QL Bytes/Percent	18	0%	0%

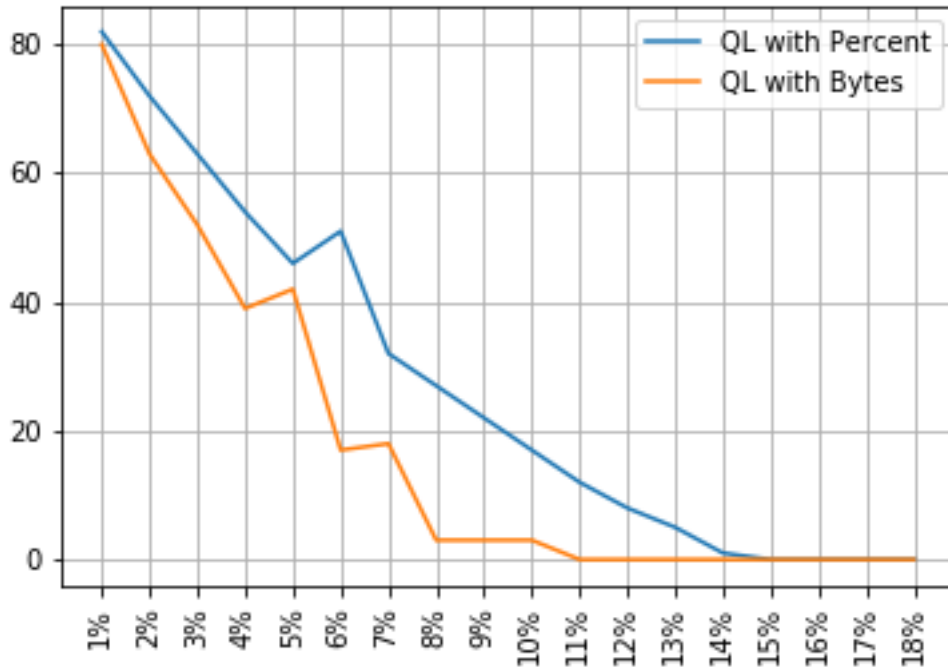
QL Percent vs bytes with 64 bytes packets



QL 퍼센트 대 200바이트 패킷 바이트

Policy-map		Drop rate with PC	Drop rate with Bytes
QL Bytes/Percent 1	1	82%	80%
QL Bytes/Percent 2	2	72%	63%
QL Bytes/Percent 3	3	63%	52%
QL Bytes/Percent 4	4	54%	39%
QL Bytes/Percent 5	5	46%	42%
QL Bytes/Percent 6	6	51%	17%
QL Bytes/Percent 7	7	32%	18%
QL Bytes/Percent 8	8	27%	3%
QL Bytes/Percent 9	9	22%	3%
QL Bytes/Percent 10	10	17%	3%
QL Bytes/Percent 11	11	12%	0%
QL Bytes/Percent 12	12	8%	0%
QL Bytes/Percent 13	13	5%	0%
QL Bytes/Percent 14	14	1%	0%
QL Bytes/Percent 15	15	0%	0%
QL Bytes/Percent 16	16	0%	0%
QL Bytes/Percent 17	17	0%	0%
QL Bytes/Percent 18	18	0%	0%

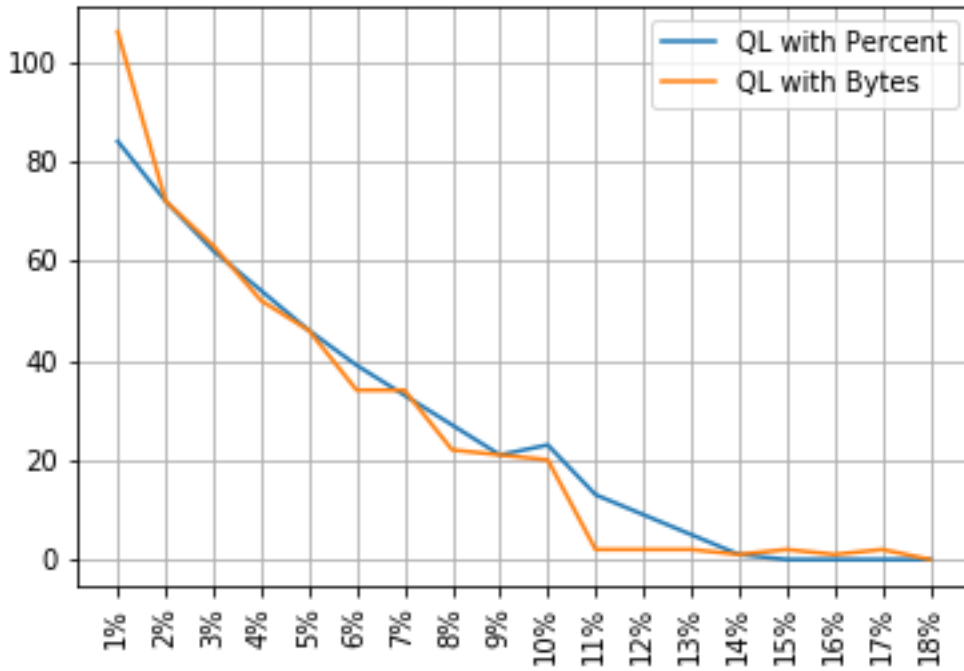
QL Percent vs bytes with 200 bytes packets



QL 퍼센트 대 300바이트 패킷 바이트

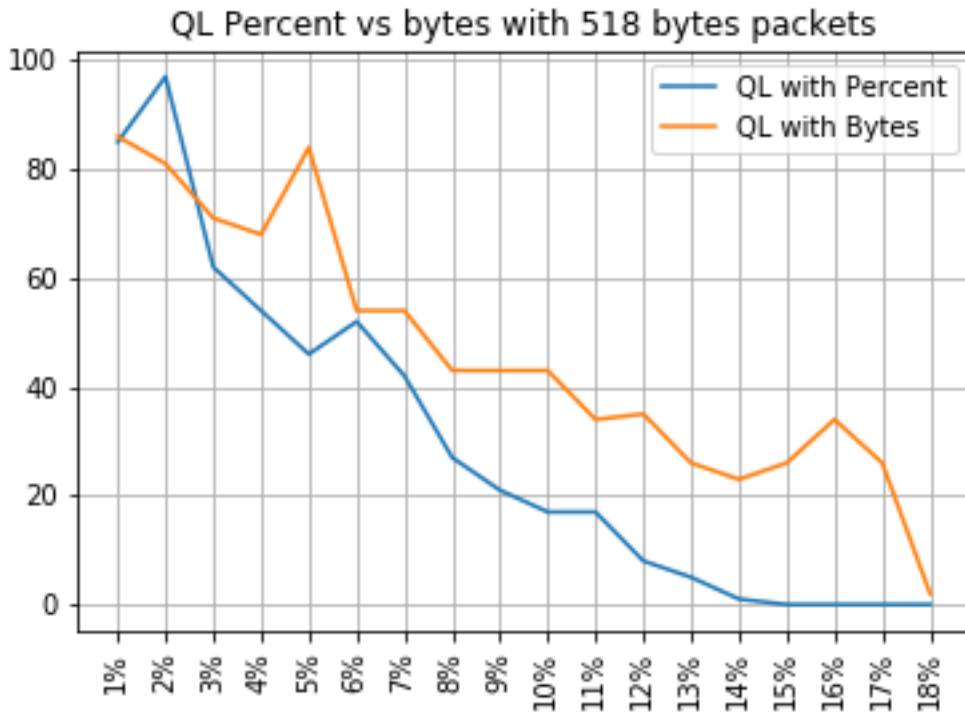
Policy-map		Drop rate with PC	Drop rate with Bytes
QL Bytes/Percent 1	1	84%	106%
QL Bytes/Percent 2	2	72%	72%
QL Bytes/Percent 3	3	62%	63%
QL Bytes/Percent 4	4	54%	52%
QL Bytes/Percent 5	5	46%	46%
QL Bytes/Percent 6	6	39%	34%
QL Bytes/Percent 7	7	33%	34%
QL Bytes/Percent 8	8	27%	22%
QL Bytes/Percent 9	9	21%	21%
QL Bytes/Percent 10	10	23%	20%
QL Bytes/Percent 11	11	13%	2%
QL Bytes/Percent 12	12	9%	2%
QL Bytes/Percent 13	13	5%	2%
QL Bytes/Percent 14	14	1%	1%
QL Bytes/Percent 15	15	0%	2%
QL Bytes/Percent 16	16	0%	1%
QL Bytes/Percent 17	17	0%	2%
QL Bytes/Percent 18	18	0%	0%

QL Percent vs bytes with 300 bytes packets



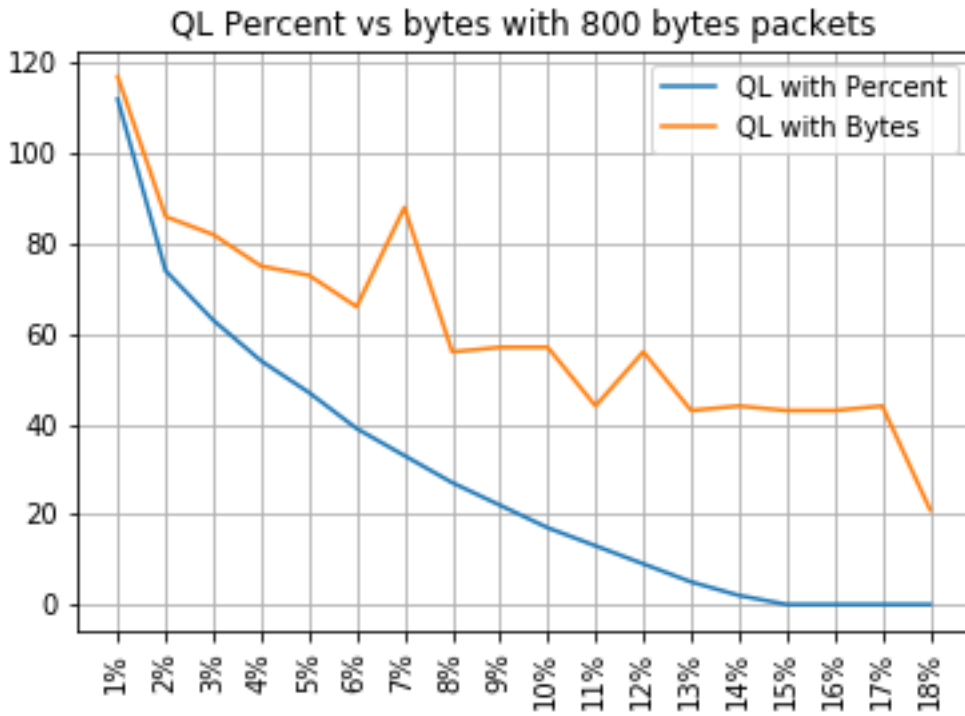
QL 퍼센트 대 518바이트 패킷 바이트

Policy-map		Drop rate with PC	Drop rate with Bytes
QL Bytes/Percent 1	1	85%	86%
QL Bytes/Percent 2	2	97%	81%
QL Bytes/Percent 3	3	62%	71%
QL Bytes/Percent 4	4	54%	68%
QL Bytes/Percent 5	5	46%	84%
QL Bytes/Percent 6	6	52%	54%
QL Bytes/Percent 7	7	42%	54%
QL Bytes/Percent 8	8	27%	43%
QL Bytes/Percent 9	9	21%	43%
QL Bytes/Percent 10	10	17%	43%
QL Bytes/Percent 11	11	17%	34%
QL Bytes/Percent 12	12	8%	35%
QL Bytes/Percent 13	13	5%	26%
QL Bytes/Percent 14	14	1%	23%
QL Bytes/Percent 15	15	0%	26%
QL Bytes/Percent 16	16	0%	34%
QL Bytes/Percent 17	17	0%	26%
QL Bytes/Percent 18	18	0%	2%



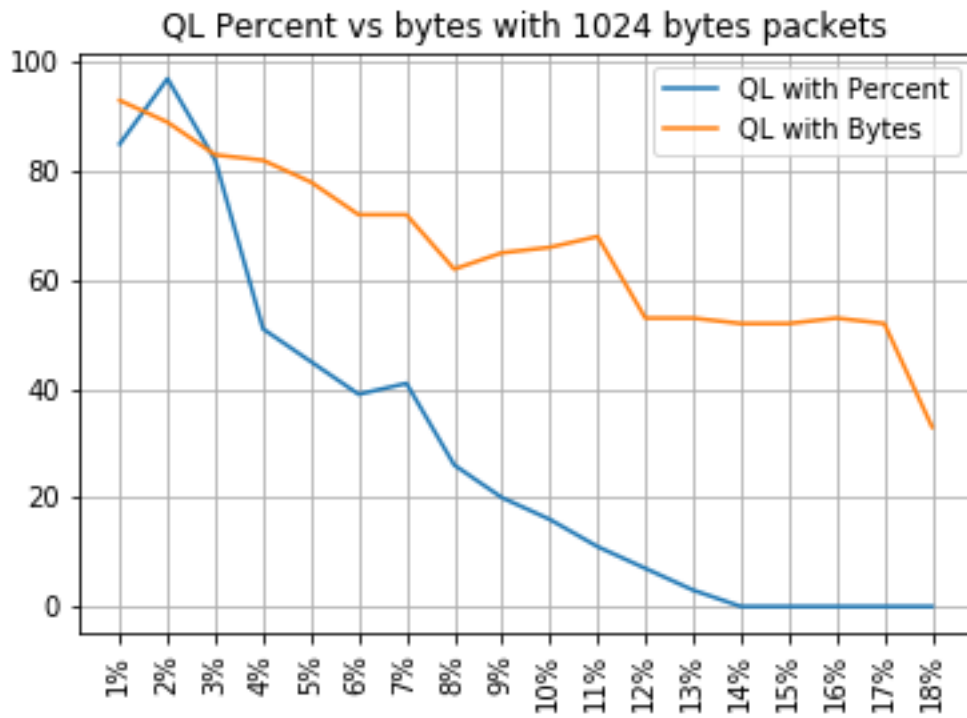
QL 퍼센트 대 800바이트 패킷 바이트

Policy-map		Drop rate with PC	Drop rate with Bytes
QL Bytes/Percent 1	1	112%	117%
QL Bytes/Percent 2	2	74%	86%
QL Bytes/Percent 3	3	63%	82%
QL Bytes/Percent 4	4	54%	75%
QL Bytes/Percent 5	5	47%	73%
QL Bytes/Percent 6	6	39%	66%
QL Bytes/Percent 7	7	33%	88%
QL Bytes/Percent 8	8	27%	56%
QL Bytes/Percent 9	9	22%	57%
QL Bytes/Percent 10	10	17%	57%
QL Bytes/Percent 11	11	13%	44%
QL Bytes/Percent 12	12	9%	56%
QL Bytes/Percent 13	13	5%	43%
QL Bytes/Percent 14	14	2%	44%
QL Bytes/Percent 15	15	0%	43%
QL Bytes/Percent 16	16	0%	43%
QL Bytes/Percent 17	17	0%	44%
QL Bytes/Percent 18	18	0%	21%



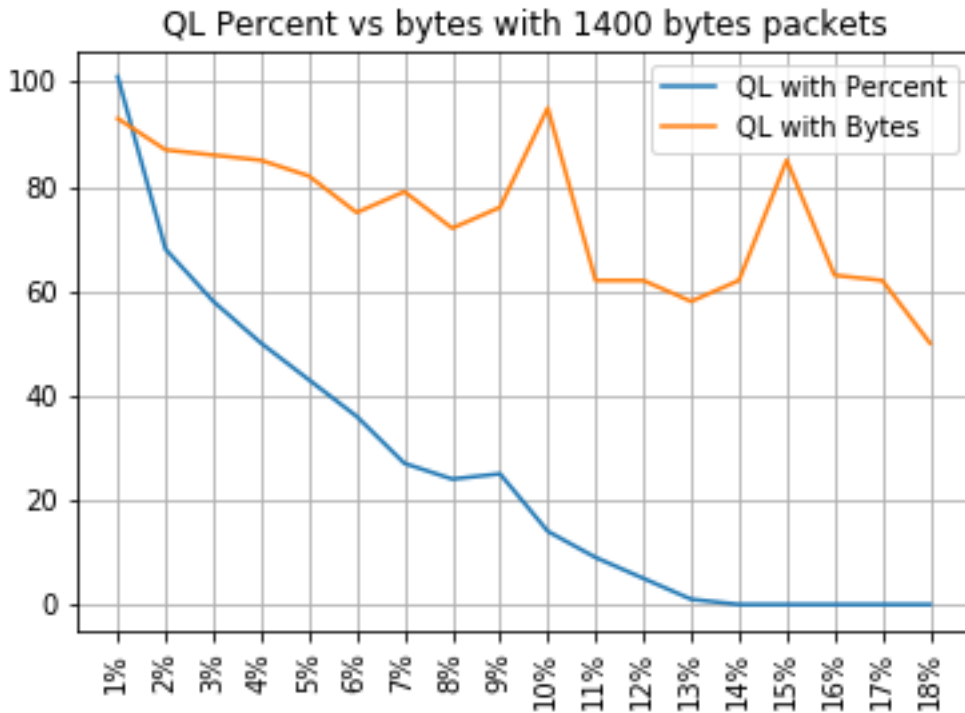
QL 퍼센트 대 1024바이트 패킷 바이트

Policy-map		Drop rate with PC	Drop rate with Bytes
QL Bytes/Percent 1	1	85%	93%
QL Bytes/Percent 2	2	97%	89%
QL Bytes/Percent 3	3	82%	83%
QL Bytes/Percent 4	4	51%	82%
QL Bytes/Percent 5	5	45%	78%
QL Bytes/Percent 6	6	39%	72%
QL Bytes/Percent 7	7	41%	72%
QL Bytes/Percent 8	8	26%	62%
QL Bytes/Percent 9	9	20%	65%
QL Bytes/Percent 10	10	16%	66%
QL Bytes/Percent 11	11	11%	68%
QL Bytes/Percent 12	12	7%	53%
QL Bytes/Percent 13	13	3%	53%
QL Bytes/Percent 14	14	0%	52%
QL Bytes/Percent 15	15	0%	52%
QL Bytes/Percent 16	16	0%	53%
QL Bytes/Percent 17	17	0%	52%
QL Bytes/Percent 18	18	0%	33%



QL 퍼센트 대 1400바이트 패킷 바이트

Policy-map	Drop rate with PC	Drop rate with Bytes
QL Bytes/Percent 1	101%	93%
QL Bytes/Percent 2	68%	87%
QL Bytes/Percent 3	58%	86%
QL Bytes/Percent 4	50%	85%
QL Bytes/Percent 5	43%	82%
QL Bytes/Percent 6	36%	75%
QL Bytes/Percent 7	27%	79%
QL Bytes/Percent 8	24%	72%
QL Bytes/Percent 9	25%	76%
QL Bytes/Percent 10	14%	95%
QL Bytes/Percent 11	9%	62%
QL Bytes/Percent 12	5%	62%
QL Bytes/Percent 13	1%	58%
QL Bytes/Percent 14	0%	62%
QL Bytes/Percent 15	0%	85%
QL Bytes/Percent 16	0%	63%
QL Bytes/Percent 17	0%	62%
QL Bytes/Percent 18	0%	50%



드롭과 패킷 크기 간의 관계

앞서 언급했듯이 ASR 920은 11.75MB의 내부 패킷 버퍼를 각 256바이트의 45898 Qnode로 분할합니다.

- 크기가 256바이트 미만인 패킷의 경우 정확히 1개의 Qnode가 사용됩니다
- 크기가 1024바이트인 패킷의 경우 4개의 Qnode가 사용됩니다
- 크기가 257바이트인 패킷의 경우 2개의 Qnode가 사용되고 사용되지 않은 255바이트는 손실됩니다

따라서 작은 패킷으로 저장할 수 있는 것보다 작은 양의 큰 패킷을 저장할 수 있습니다. 동등한 제한 크기의 삭제 속도와 패킷 크기 사이의 관계가 필요합니다.

동등한 바이트와 백분율 값의 차이

설명한 것처럼 11.75MB 공유 버퍼에는 45898 개의 Qnode 가 있으며, 계산의 용이성을 위해 45900으로 반올림됩니다.

queue-limit 백분율 계산은 11.75MB의 백분율이 아니라 45900 Qnode의 백분율을 계산합니다. queue-limit percent 10은 45900의 10%를 의미하며 4590개의 Qnode를 제공합니다. 또한 할당된 Qnode의 백분율은 해당 크기와 독립적으로 큐에 저장할 수 있는 패킷의 수로 간주됩니다. 앞의 예제로 돌아가보면 다음과 같습니다.

queue-limit 백분율 10 = 4590 Qnodes = 4590 패킷.

이 계산은 패킷 크기와 무관하므로 256바이트 이하의 패킷에 대해서는 하나의 Qnode만 실제로 사용되며 Qnode와 패킷 간의 등가는 유지됩니다.

queue-limit 백분율 10 = 4590 Qnodes = 4590 패킷(256바이트) = 4590*256바이트 = 1.175MB = 버퍼의 10%

그러나 패킷이 클수록 더 많은 양의 버퍼가 할당됩니다. 예를 들어, 각 패킷이 4개의 Qnode를 사용하는 경우 1024바이트 패킷에 대한 계산입니다.

queue-limit 백분율 10 = 4590 Qnodes = 4590 패킷 1024바이트 = 4590*4*1024바이트 = 18.8MB
= 40% 버퍼

주의: queue-limit 백분율의 높은 값은 구성하지 않는 것이 좋습니다.

queue-limit 백분율의 높은 값을 구성할 경우, 단일 인터페이스가 일시적으로 11.75MB의 모든 공유 버퍼를 차지할 수 있습니다.

결론

- 큐 제한 바이트의 효율성이 작은 패킷에서 더 좋다는 것을 분명히 알 수 있습니다. 큐 제한 바이트 <x>가 최대 300바이트의 큐 제한 <x>보다 더 효과적입니다
- 300바이트 패킷에서는 queue-limit 바이트 및 queue-limit 백분율 효율이 동일합니다
- 300바이트 이상의 패킷 크기인 queue-limit 퍼센트가 더 효율적입니다. 인터넷 트래픽은 평균 518바이트이므로, 고객이 보고한 것처럼 실제 시나리오에서는 queue-limit 퍼센트가 더 큰 이득을 가져다 줍니다
- queue-limit 백분율의 효율성은 패킷 크기와 관련하여 선형적으로 개선됩니다(패킷이 클수록 queue-limit 백분율이 queue-limit 바이트에 비해 더 효율적임)
- Queue-limit 백분율은 256MB 이상의 패킷에 대한 버퍼 공간 할당에서 더 많이 사용됩니다

실제 트래픽 시나리오에서 백분율 값을 근사화하는 방법

크기가 256바이트이고 queue-limit이 10%인 패킷이 있는 경우, 다음과 같은 등가물이 유효함을 이미 알고 있습니다.

minimum queue-limit = 4590 Qnodes = 4590*256 bytes = 1.175 MB = 10% of the buffer

512바이트 패킷에서는 사용량만 2배이고, 1024바이트 패킷에서는 4배입니다.

즉, 실제 queue-limit은 버퍼의 최소 10%이며, 최대 MTU를 1500바이트로 가정할 경우 단일 패킷을 저장하려면 6개의 Qnode가 필요합니다. 그러면 최대 queue-limit은 다음과 같습니다.

maximum queue-limit = 4590*6 Qnodes = 4590*256*6 bytes = 7.05 MB = 60% of the buffer

이 방법으로 queue-limit percent 10을 사용하여 버퍼 사용량의 하한 및 상한을 정의할 수 있으므로 일반적으로 평균 최대 버퍼 사용량은 대략 다음과 같습니다.

`ceil(avg_pkt_size/256)*((qlimit_percent/45900)*100)`

랩 장비의 예:

```
GigabitEthernet0/0/1 is up, line protocol is up
Hardware is 24xGE-4x10GE-FIXED-S, address is 70df.2f2f.ed01 (bia 70df.2f2f.ed01)
Internet address is 10.12.10.47/31
MTU 8900 bytes, BW 1000000 Kbit/sec, DLY 10 usec,
reliability 255/255, txload 25/255, rxload 30/255
Encapsulation ARPA, loopback not set
Keepalive set (10 sec)
```

```

Full Duplex, 1000Mbps, link type is auto, media type is T
output flow-control is unsupported, input flow-control is on
Carrier delay is 0 msec
ARP type: ARPA, ARP Timeout 04:00:00
Last input 00:00:00, output 00:00:01, output hang never
Last clearing of "show interface" counters 00:11:43
Input queue: 0/375/0/0 (size/max/drops/flushes); Total output drops: 2036062
Queueing strategy: Class-based queueing
Output queue: 0/40 (size/max)
30 second input rate 118520000 bits/sec, 18902 packets/sec
30 second output rate 101646000 bits/sec, 16124 packets/sec
13185272 packets input, 10328798549 bytes, 0 no buffer
Received 0 broadcasts (0 IP multicasts)
0 runts, 0 giants, 0 throttles
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
0 watchdog, 235 multicast, 0 pause input
11247114 packets output, 8870166880 bytes, 0 underruns <<< avg_pkt_size = 8870166880/11247114 =
788.66 bytes
0 output errors, 0 collisions, 0 interface resets
0 unknown protocol drops
0 babbles, 0 late collision, 0 deferred
0 lost carrier, 0 no carrier, 0 pause output
0 output buffer failures, 0 output buffers swapped out
avg_pkt_size는 8870166880/11247114 ~ 788바이트로 계산할 수 있습니다.

```

queue-limit percent 10의 평균 최대 버퍼 사용량은 다음과 같습니다.

```
ceil(avg_pkt_size/256)*((45900/100)*qlimit_percent)
```

Calculation example with Python:

```

>>>import math
>>>math.ceil(788/256)*((45900/100)*10)
18360.0

```

=> 18360 Qnodes = 18360 * 256 bytes = 4.7 MB = 40% of the buffer

마이크로버스트 버퍼 사용량을 확인하는 방법

16.9.3 이전 Cisco IOS-XE 릴리스에서는 인터페이스의 공유 버퍼가 데이터 및 제어 패킷(예: BFD, 라우팅 프로토콜, ARP, LDP, punt keepalive)에 사용되었습니다. 인스턴트 버퍼 사용량을 확인하려면 다음 명령을 사용할 수 있습니다.

```
ASR-920-1#request platform software sdcli "nile bm reg buffertablefreelistcount show 0 0 0"
```

16.9.3 이후 일부 변경 사항은 버퍼 사용량을 개선하기 위해 도입되었으며 2개로 분할되었습니다. 1024개 항목(256KB)은 제어 트래픽용으로 예약되었으며 나머지는 데이터 트래픽용으로 예약되었습니다.

이 경우 버퍼 사용량은 다음 명령으로 모니터링할 수 있습니다.

```

ASR-920-1#request platform software sdcli "nile bm reg
supervisorresourcesreservedcounttableaccess sh 0 0 0"
reservedUsedCount = 48 (0x30)
reservedFreeCount = 976 (0x3d0)

```

```

ASR-920-1#request platform software sdcli "nile bm reg
supervisorresourcesreservedcounttableaccess sh 0 2 0"

```



```
reservedUsedCount = 8114 (0x1fb2)
reservedFreeCount = 37784 (0x9398)
```

버퍼에서 마이크로버스트를 처리하는 경우 reservedUsedCount 값이 0과 다른 것을 보려면 명령을 여러 번 반복해야 합니다.

버퍼 사용량은 reservedUsedCount/reservedFreeCount로 간단하게 계산할 수 있습니다(예: 8114/37784 = 21,5% 사용됨). 버스트가 끝나면 버퍼가 0으로 빠르게 떨어지거나 가까워져야 합니다.

Cisco IOS-XE 릴리스 17.6.1에서는 데이터 및 제어 트래픽(네트워크에서 제어 트래픽 비율이 높은 경우 권장) 모두에 대해 전체 버퍼를 사용하거나 앞에서 설명한 대로 버퍼를 2로 분할하도록 선택할 수 있습니다. 이 명령의 컨피그레이션으로 선택합니다(기본적으로 비활성화됨).

```
ACDC-920-1(config)#platform qos-buffer enhance enable
ACDC-920-1(config)#no platform qos-buffer enhance enable
```

Cisco IOS-XE 릴리스 17.7.1에서는 제어 트래픽에 할당할 크기를 선택할 수도 있습니다.

```
ACDC-920-1(config)#platform qos-buffer enhance [1-4]
```

여기서

- 1은 256KB의 제어 버퍼를 나타냅니다.
- 2는 500KB를 나타냅니다.
- 3은 756KB를 나타냄
- 4는 1MB를 나타냄

이 번역에 관하여

Cisco는 전 세계 사용자에게 다양한 언어로 지원 콘텐츠를 제공하기 위해 기계 번역 기술과 수작업 번역을 병행하여 이 문서를 번역했습니다. 아무리 품질이 높은 기계 번역이라도 전문 번역가의 번역 결과물만큼 정확하지는 않습니다. Cisco Systems, Inc.는 이 같은 번역에 대해 어떠한 책임도 지지 않으며 항상 원본 영문 문서(링크 제공됨)를 참조할 것을 권장합니다.