

Come gestire i microburst in ASR 920, differenze tra byte e percentuale con limite di coda

Sommario

[Introduzione](#)

[Prerequisiti](#)

[Requisiti](#)

[Componenti usati](#)

[Premesse](#)

[Configura limite coda](#)

[Differenze con RSP3 e NCS 520](#)

[Test Lab QL \(Queue-Limit\), differenze tra byte e percentuale](#)

[Ambito e parametri del test](#)

[Procedura di prova](#)

[Esempio](#)

[Risultati dei test di laboratorio](#)

[Percentuale SQL rispetto ai byte con pacchetti da 64 byte](#)

[Percentuale SQL rispetto ai byte con pacchetti da 200 byte](#)

[Percentuale SQL rispetto ai byte con pacchetti da 300 byte](#)

[Percentuale SQL rispetto ai byte con pacchetti da 518 byte](#)

[Percentuale SQL rispetto ai byte con pacchetti da 800 byte](#)

[Percentuale SQL rispetto ai byte con pacchetti da 1024 byte](#)

[Percentuale SQL rispetto ai byte con pacchetti da 1400 byte](#)

[Relazione tra le gocce e le dimensioni del pacchetto](#)

[Differenza tra byte equivalenti e valori percentuali](#)

[Conclusioni](#)

[Come approssimare i valori percentuali in scenari di traffico reale](#)

[Verifica dell'utilizzo del buffer Microburst](#)

Introduzione

Questo documento descrive come gestire i microburst nei router ASR 920, che sono spesso la causa delle perdite di pacchetti di output dell'interfaccia. Nel dettaglio viene mostrata la differenza tra l'uso dei byte e la percentuale con il comando queue-limit.

Prerequisiti

Requisiti

Cisco raccomanda la conoscenza dei seguenti argomenti:

ASR serie 920 router

Componenti usati

Per questo documento, è stato usato un router ASR 920 con software Cisco IOS-XE 16.9.6.

IXIA viene utilizzato come generatore di traffico del test di laboratorio.

Le informazioni discusse in questo documento fanno riferimento a dispositivi usati in uno specifico ambiente di emulazione. Su tutti i dispositivi menzionati nel documento la configurazione è stata ripristinata ai valori predefiniti. Se la rete è operativa, valutare attentamente eventuali conseguenze derivanti dall'uso dei comandi.

Premesse

I microburst vengono definiti piccoli picchi nel traffico di rete. Nelle reti di accesso per provider di servizi, ad esempio, questo si verifica in genere in scenari di mancata corrispondenza della velocità in cui un flusso di traffico entra nel router da un'interfaccia ad alta velocità, come 10 Gigabit Ethernet (GE), ed esce da un'interfaccia a bassa velocità, come 1 GE.

Il problema più comune che i microbit causano nei router ASR 920 è che i pacchetti dell'interfaccia vengono scartati nella direzione di output. Questo si verifica quando sull'interfaccia in ingresso sono presenti burst superiori alla frequenza dell'interfaccia in uscita, per un periodo di tempo molto breve (ordine di millisecondi). Durante questo periodo, i pacchetti devono essere memorizzati nel buffer.

Sulle piattaforme ASR 920 / RSP2, il buffer predefinito allocato per le code su ciascuna interfaccia 1 GE è 48 KB, mentre per le code su ciascuna interfaccia 10 GE è 120 KB.

Inoltre, è disponibile un buffer condiviso di 11,75 MB. Quando il buffer dell'interfaccia predefinita non è sufficiente per contenere una frammentazione di pacchetti, è possibile utilizzare il buffer condiviso.

Per abilitare l'utilizzo del buffer condiviso per un'interfaccia, è necessario configurare un criterio QoS nell'interfaccia che definisca il parametro `queue-limit`.

Configura limite coda

Questo è un esempio di configurazione del limite di coda per l'interfaccia GE 0/0/1:

```
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
```

Questa policy non corrisponde ad alcun traffico, quindi non ha alcun impatto. Il criterio consente solo di aumentare il buffer della coda dell'interfaccia GE 0/0/1.

Il comando **match qos-group 99** è necessario perché non è supportato per configurare queue-limit in una classe non foglia, quindi è necessario utilizzare una class-map fittizia come classe padre per configurare il comando queue-limit nella classe foglia-default:

```
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
```

Il limite di coda può essere impostato in diversi modi:

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

Il parametro e il valore devono essere selezionati in base ai requisiti di rete.

Per comprendere tali parametri e il modo in cui una modifica dei valori influisce sull'utilizzo del buffer, è possibile fare riferimento al test di laboratorio illustrato nella sezione successiva.

Nota: la configurazione di un criterio QoS nelle interfacce del canale della porta non è supportata solo nelle interfacce fisiche del canale della porta

```
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
```

Differenze con RSP3 e NCS 520

[Funzionalità QoS del modulo Cisco RSP3:](#)

- Il modulo RSP3 dispone di 4 GB di buffer di pacchetti esterni per NPU
- Il modulo RSP3 supporta 4800 code
- Per impostazione predefinita, il modulo RSP3 supporta un limite di 1 MB per coda
- La percentuale di limite della coda è considerata su 1 GB dei buffer totali

Per i router con supervisor RSP3 e NCS 520, il numero di gruppi qos configurabili è limitato a 0-7:

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

Per impostazione predefinita, in NCS 520 è presente un buffer di coda da 2 MB condiviso tra tutte le interfacce. Un buffer esterno da 2 GB è accessibile quando è configurata una mappa dei criteri con limite di coda. C'è anche una differenza nei byte e nei parametri us per il queue-limit:

```
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
```

Test Lab QL (Queue-Limit), differenze tra byte e percentuale

Ambito e parametri del test

Come osservato in precedenza, il valore massimo configurabile per il limite della coda in byte è 2097152, che è circa il 18% del buffer condiviso sulle piattaforme ASR 920 (11,75 MB ~ 45898 * 256 byte).

Tuttavia, se si configura il limite della coda in percentuale, è possibile passare al 100%. Pertanto, per confrontare percentuale e byte con valori equivalenti, il test accetta valori di byte da 117498 a 2097152 byte e valori di percentuale limite di coda dall'1% al 18%:

```
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
```

Sono configurate 36 mappe delle policy: 18 con valori limite di coda compresi tra 1% e 18% e gli altri 18 con valori limite di coda compresi tra 117498 e 2097152 byte.

```
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

Ogni policy viene testata sullo stesso traffico di microburst, generato con IXIA. Il traffico arriva su una porta 10 GE dell'ASR 920 ed esce su una porta 1 GE dello stesso router.

I burst sono costituiti da 2000 pacchetti a 4 Gb/s a intervalli di 5 secondi. Questa è la durata della frammentazione in base a una dimensione pacchetto specifica:

```
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
```

Procedura di prova

Passaggio 1. Applicare la mappa dei criteri QUEUE-LIMIT-BYTES-X (X=1) con i byte Y=117498 nell'interfaccia in uscita.

Passaggio 2. Eseguire il traffico microburst per 1 minuto.

Passaggio 3. Misurare il numero totale di pacchetti in output e il numero di pacchetti scartati.

Passaggio 4. Calcolare il rapporto tra i pacchetti ignorati e il totale dei pacchetti di output.

Passaggio 5. Ripetere una volta dal passaggio 1, questa volta utilizzare la mappa dei criteri QUEUE-LIMIT-PERCENT-X con la percentuale di limite della coda X, dove X=1.

Passaggio 6. Ripetere dal passaggio 1 con X=X+1 per il nome della mappa dei criteri e il valore percentuale, Y=Y+117498 byte. Ripetere fino a X=18 e Y=2097152.

Esempio

Misurazione con la mappa dei criteri QUEUE-LIMIT-BYTES-1:

```
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
```

Misurazione dopo 1 minuto di traffico di microburst:

```
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

```

Delta perdite pacchetti: $684561562 - 684263427 = 298135$

Delta di output pacchetti totali: $7829753878 - 7829367523 = 386355$

Rapporto tra i pacchetti ignorati e quelli in uscita: $298135 / 386355 = 77\%$

Risultati dei test di laboratorio

Come spiegato, vengono testate 36 mappe delle politiche: 18 configurati con valori limite di coda compresi tra l'1% e il 18% vengono confrontati con gli altri 18 criteri configurati con valori limite di coda compresi tra 117498 e 2097152 byte. Ogni mappa delle policy viene testata sullo stesso traffico di microburst generato con IXIA.

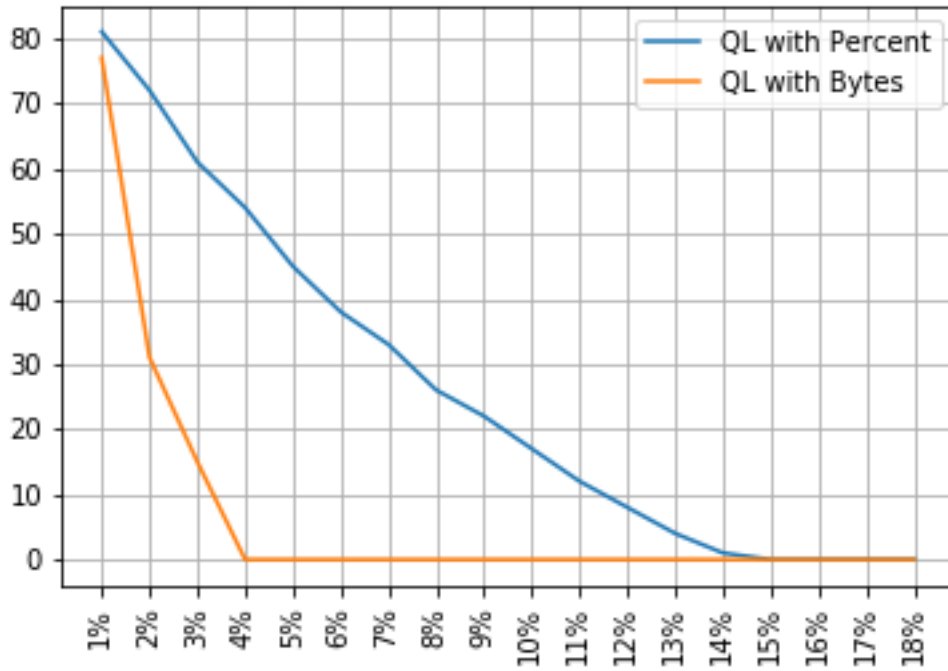
I risultati di questa prova, ripetuti 7 volte in questa sezione, permettono di verificare i risultati con dimensioni di pacchetto diverse, in byte: 64, 200, 300, 518, 800, 1024, 1400.

Per facilitare la lettura, per ogni dimensione del pacchetto i risultati vengono esposti in una tabella e poi rappresentati graficamente.

Percentuale SQL rispetto ai byte con pacchetti da 64 byte

| 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% |

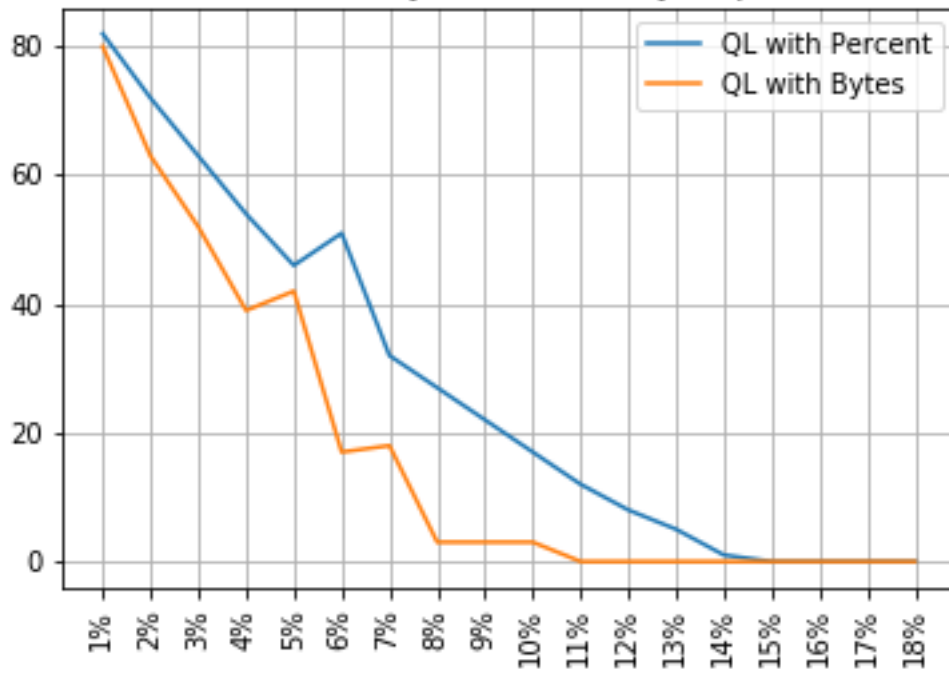
QL Percent vs bytes with 64 bytes packets



Percentuale SQL rispetto ai byte con pacchetti da 200 byte

| 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% |

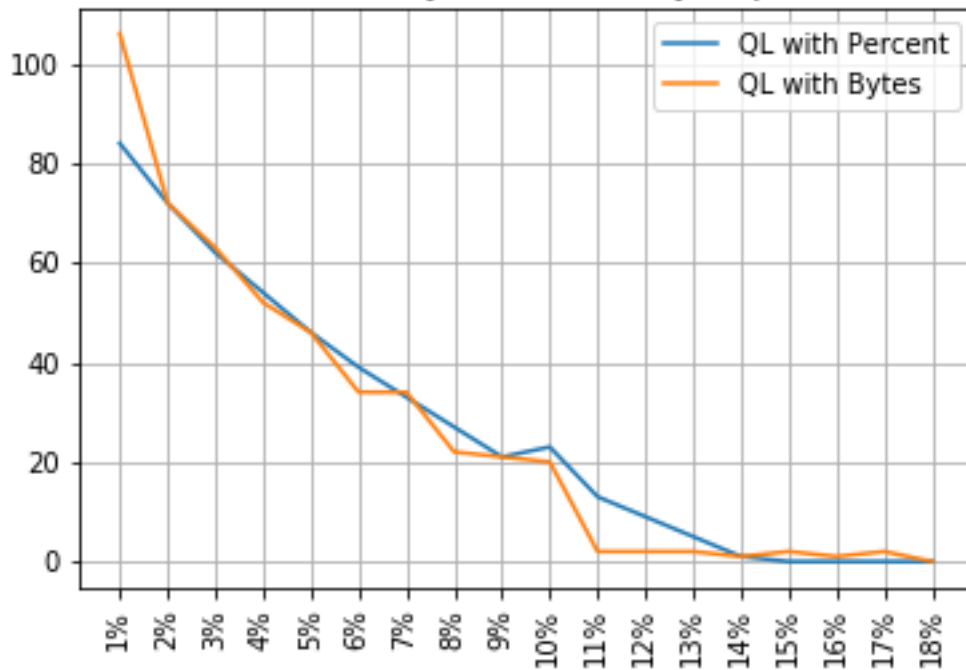
QL Percent vs bytes with 200 bytes packets



Percentuale SQL rispetto ai byte con pacchetti da 300 byte

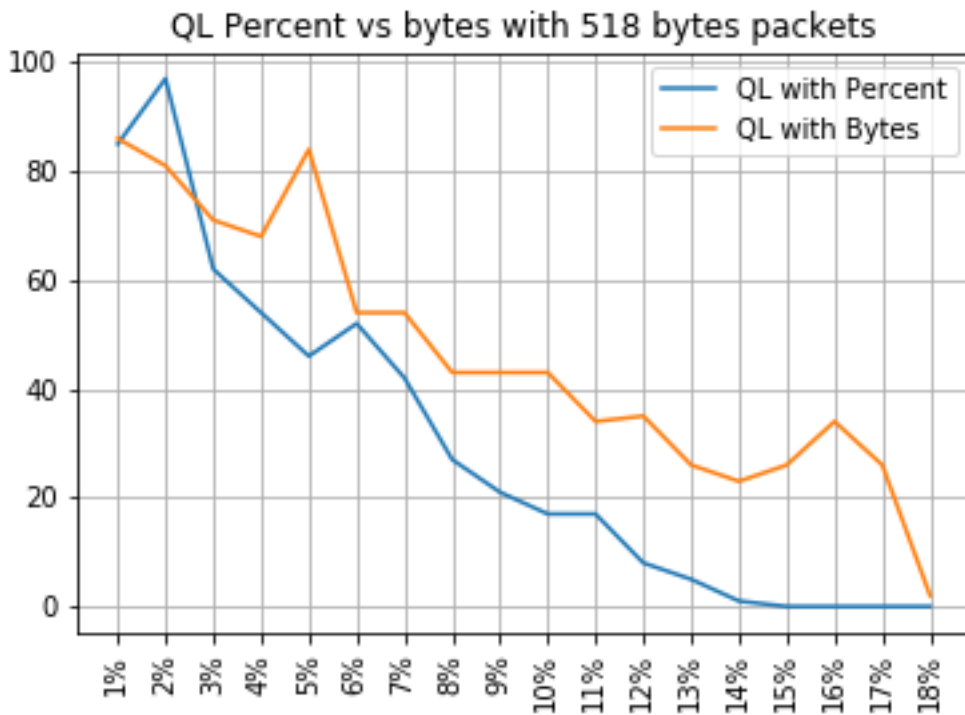
| Policy-map | Drop rate with PC | Drop rate with Bytes |
|---------------------|-------------------|----------------------|
| QL Bytes/Percent 1 | 84% | 106% |
| QL Bytes/Percent 2 | 72% | 72% |
| QL Bytes/Percent 3 | 62% | 63% |
| QL Bytes/Percent 4 | 54% | 52% |
| QL Bytes/Percent 5 | 46% | 46% |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

QL Percent vs bytes with 300 bytes packets



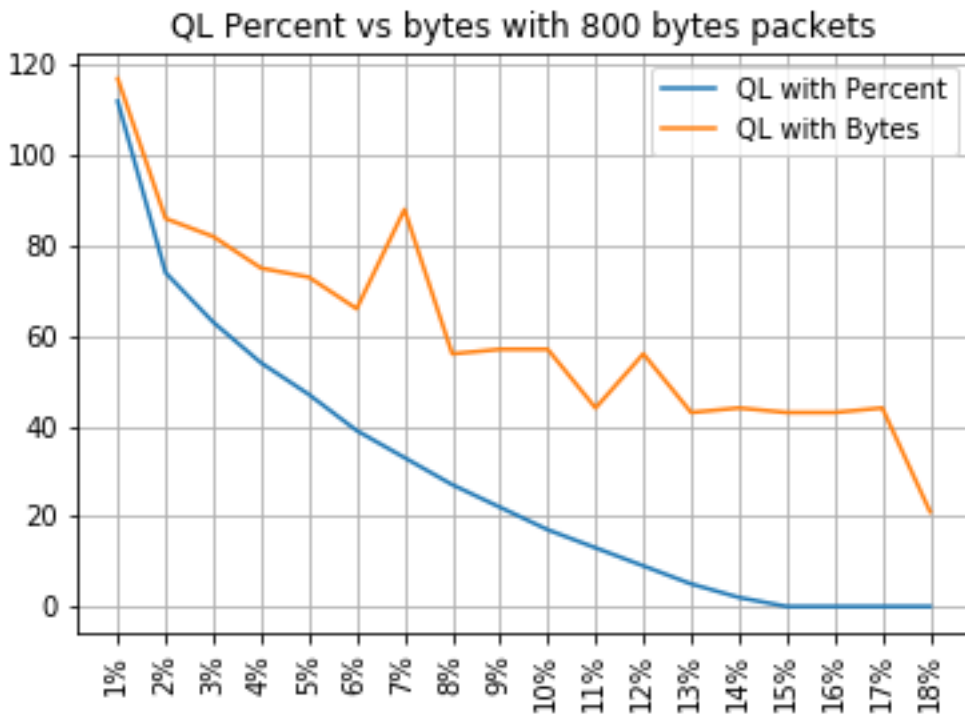
Percentuale SQL rispetto ai byte con pacchetti da 518 byte

| 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% |



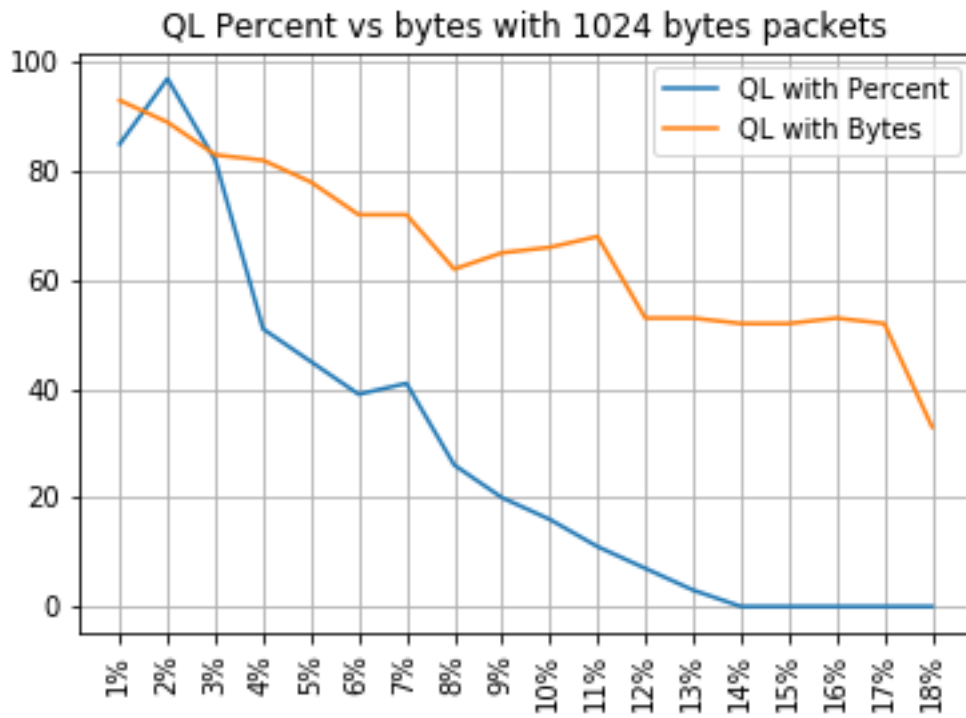
Percentuale SQL rispetto ai byte con pacchetti da 800 byte

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



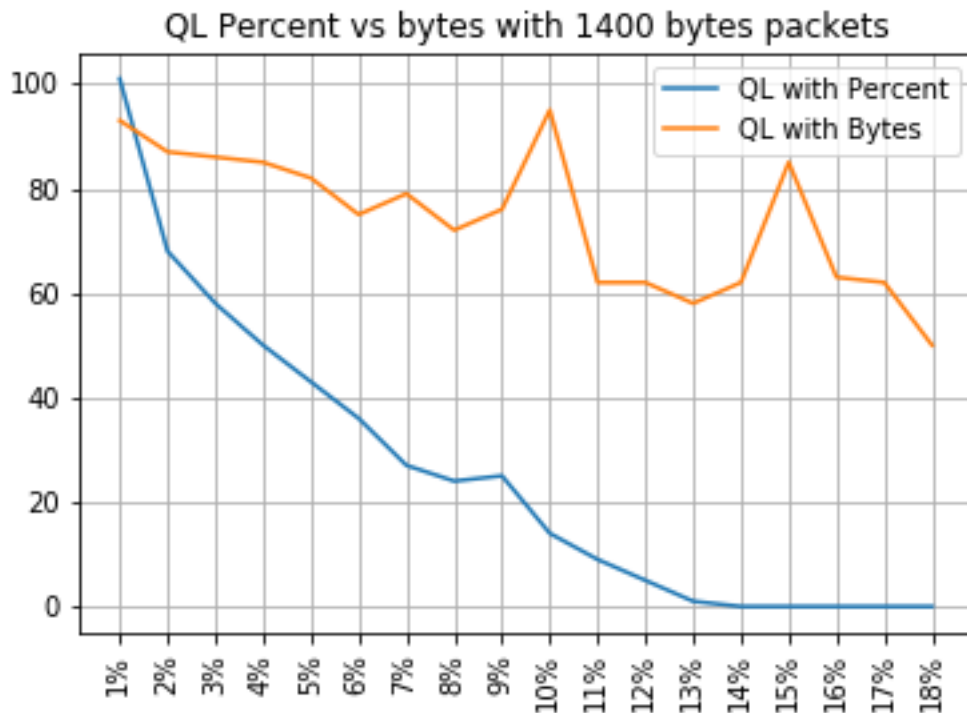
Percentuale SQL rispetto ai byte con pacchetti da 1024 byte

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



Percentuale SQL rispetto ai byte con pacchetti da 1400 byte

| 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% |



Relazione tra le gocce e le dimensioni del pacchetto

Come accennato, l'ASR 920 ha un buffer di pacchetto interno di 11,75 MB diviso in 45898 nodi di 256 byte ciascuno.

- Per un pacchetto con dimensioni < 256 byte, viene utilizzato esattamente 1 Qnode
- Per un pacchetto con dimensioni di 1024 byte, vengono utilizzati 4 nodi QL
- Per un pacchetto con dimensioni di 257 byte, vengono utilizzati 2 Qnode e i 255 byte inutilizzati vengono persi

È quindi possibile memorizzare una quantità di pacchetti di grandi dimensioni inferiore a quella che è possibile memorizzare con pacchetti di piccole dimensioni. È prevista la relazione tra la velocità di rilascio e le dimensioni del pacchetto a limiti di coda equivalenti.

Differenza tra byte equivalenti e valori percentuali

Come spiegato, ci sono 45898 nodi Q in 11,75 MB di buffer condiviso, arrotondati a 45900 per facilitare il calcolo.

Il calcolo della percentuale del limite di coda non calcola la percentuale degli 11,75 MB, ma quella degli 45900 nodi QoS. Quindi, la percentuale di limite di coda 10 significa 10% di 45900 che fornisce 4590 nodi.

Inoltre, la percentuale di nodi allocati è considerata come il numero di pacchetti che possono essere memorizzati nella coda, indipendentemente dalle sue dimensioni. Tornando all'esempio precedente, ciò significa che:

percentuale limite coda 10 = 4590 nodi = 4590 pacchetti.

Poiché questo calcolo è indipendente dalle dimensioni del pacchetto, per pacchetti di dimensioni uguali o inferiori a 256 byte viene effettivamente utilizzato un solo Qnode e viene mantenuta l'equivalenza tra Qnode e packet:

percentuale limite di coda 10 = 4590 nodi = 4590 pacchetti di 256 byte = 4590*256 byte = 1,175 MB = 10% del buffer

Con i pacchetti più grandi, tuttavia, viene allocata una parte più generosa del buffer. Ad esempio, questo è il calcolo per i pacchetti da 1024 byte, in cui ogni pacchetto consuma 4 nodi:

percentuale limite coda 10 = 4590 nodi = 4590 pacchetti di 1024 byte = 4590*4*256 byte = 4,7 MB = 40% del buffer

Attenzione: non è consigliabile configurare valori elevati della percentuale di limite di coda.

Se si configurano valori elevati per la percentuale di limite di coda, una singola interfaccia potrebbe occupare temporaneamente tutto il buffer condiviso di 11,75 MB.

Conclusioni

- È possibile notare chiaramente che l'efficienza dei byte di limite di coda è migliore con i pacchetti di piccole dimensioni: i byte di limite di coda <x> funzionano meglio del limite di coda <x> fino a 300 byte
- A 300 byte, l'efficienza dei byte di limite di coda e della percentuale di limite di coda è la stessa
- Dimensioni dei pacchetti superiori a 300 byte, percentuale di limite della coda, sono più efficienti. Poiché il traffico Internet è pari alla media di 518 byte, gli scenari reali traggono maggiore vantaggio dalla percentuale di limite di coda, come riportato dai clienti
- L'efficienza della percentuale di limite di coda migliora in modo lineare in base alle dimensioni del pacchetto (più grandi sono i pacchetti, maggiore è la percentuale di limite di coda efficiente rispetto ai byte di limite di coda)
- La percentuale di limite di coda è implementata per essere più generosa nell'allocazione dello spazio del buffer per i pacchetti di dimensioni superiori a 256 MB

Come approssimare i valori percentuali in scenari di traffico reale

Se si hanno pacchetti con dimensioni di 256 byte e un limite di coda del 10%, l'equivalenza è già nota:

$\text{minimum queue-limit} = 4590 \text{ Qnodes} = 4590 * 256 \text{ bytes} = 1.175 \text{ MB} = 10\% \text{ of the buffer}$

Con pacchetti da 512 byte solo l'uso è il doppio, con pacchetti da 1024 byte solo è quattro volte tanto, e così via.

Ciò significa che il limite di coda effettivo è almeno il 10% del buffer e, se si ipotizza una MTU massima di 1500 byte, sono necessari 6 nodi QI per archiviare un singolo pacchetto, il che fornisce un limite di coda massimo di:

$\text{maximum queue-limit} = 4590 * 6 \text{ Qnodes} = 4590 * 256 * 6 \text{ bytes} = 7.05 \text{ MB} = 60\% \text{ of the buffer}$

In questo modo è possibile definire i limiti superiore e inferiore dell'uso del buffer con una percentuale di limite di coda pari a 10, quindi più in generale l'uso medio massimo del buffer è approssimativamente:

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

Esempio da un'apparecchiatura di laboratorio:

```
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
```

È possibile calcolare la dimensione media di 8870166880/11247114 ~ 788 byte.

Con una percentuale di limite di coda pari a 10, l'utilizzo medio massimo del buffer è pari a:

```
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

Verifica dell'utilizzo del buffer Microburst

Nelle versioni Cisco IOS-XE precedenti alla 16.9.3, il buffer condiviso delle interfacce era usato sia per i pacchetti di dati che per quelli di controllo (come BFD, protocolli di routing, ARP, LDP, punt keepalive). Per verificare l'utilizzo immediato del buffer, è possibile utilizzare questo comando:

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

Dopo la versione 16.9.3 sono state introdotte alcune modifiche per migliorare l'utilizzo del buffer, che è stato suddiviso in 2: 1.024 voci (256 KB) sono state riservate al traffico di controllo e le altre

voci sono riservate al traffico di dati.

In questo caso, l'utilizzo del buffer può essere monitorato con i seguenti comandi:

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

```
ASR-920-1#request platform software sdcli "nile bm reg
supervisorresourcereservedcounttableaccess sh 0 2 0"
reservedUsedCount = 8114 (0x1fb2)
reservedFreeCount = 37784 (0x9398)
```

Tenere presente che, poiché il buffer gestisce i microburst, è necessario ripetere il comando più volte per visualizzare il valore ReservedUsedCount diverso da 0.

L'utilizzo del buffer può essere calcolato semplicemente con $\text{reserveUsedCount}/\text{reserveFreeCount}$, ad esempio $8114/37784 = 21,5\%$ utilizzato. Una volta terminata la frammentazione, il buffer deve tornare rapidamente a 0 o quasi.

Da Cisco IOS-XE versione 17.6.1, è possibile scegliere di utilizzare l'intero buffer sia per i dati che per il traffico di controllo (preferibilmente se la rete ha una velocità elevata di controllo del traffico) o suddividere il buffer in 2 come descritto in precedenza. La scelta viene effettuata con la configurazione di questa istruzione (disabilitata per impostazione predefinita):

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

Da Cisco IOS-XE versione 17.7.1, è anche possibile scegliere le dimensioni da assegnare al traffico di controllo:

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

Dove:

- 1 indica un buffer di controllo di 256 KB
- 2 indica 500 KB
- 3 indica 756 KB
- 4 indica 1 MB

Informazioni su questa traduzione

Cisco ha tradotto questo documento utilizzando una combinazione di tecnologie automatiche e umane per offrire ai nostri utenti in tutto il mondo contenuti di supporto nella propria lingua. Si noti che anche la migliore traduzione automatica non sarà mai accurata come quella fornita da un traduttore professionista. Cisco Systems, Inc. non si assume alcuna responsabilità per l'accuratezza di queste traduzioni e consiglia di consultare sempre il documento originale in inglese (disponibile al link fornito).