

ASR 920でのマイクロバーストの処理方法、キュー制限によるバイトとパーセントの違い

内容

[概要](#)

[前提条件](#)

[要件](#)

[使用するコンポーネント](#)

[背景説明](#)

[キュー制限の設定](#)

[RSP3およびNCS 520との違い](#)

[キュー制限\(QL\)ラボテスト、バイトとパーセントの違い](#)

[テストスコープとパラメータ](#)

[テスト手順](#)

[例](#)

[ラボテストの結果](#)

[64バイトパケットのQLの割合とバイトの比較](#)

[200バイトパケットのQLパーセント対バイト](#)

[300バイトのパケットでのQLの割合とバイト](#)

[518バイトパケットのQLパーセント対バイト](#)

[800バイトのパケットでのQLの割合とバイト](#)

[1024バイトパケットのQLパーセント対バイト](#)

[1400バイトパケットのQLパーセント対バイト](#)

[ドロップとパケットサイズの関係](#)

[等価バイト値とパーセント値の差](#)

[まとめ](#)

[実際のトラフィックシナリオにおけるパーセント値の近似方法](#)

[マイクロバーストバッファの使用状況を確認する方法](#)

概要

このドキュメントでは、ASR 920ルータでマイクロバーストを処理する方法について説明します。マイクロバーストは、インターフェイスの出力パケットのドロップの原因となることが多いものです。queue-limitコマンドを使用すると、バイトの使用量とパーセントの違いが詳細に表示されます。

前提条件

要件

次の項目に関する知識があることが推奨されます。

使用するコンポーネント

このドキュメントの情報は、ソフトウェアバージョンCisco IOS-XE 16.9.6が稼働するASR 920ルータに基づくものです。

IXIAは、ラボテストのトラフィックジェネレータとして使用されます。

このドキュメントの情報は、特定のラボ環境にあるデバイスに基づいて作成されました。このドキュメントで使用するすべてのデバイスは、初期（デフォルト）設定の状態から起動しています。本稼働中のネットワークでは、各コマンドによって起こる可能性がある影響を十分確認してください。

背景説明

マイクロバーストは、ネットワークトラフィックの小さな急増と呼ばれます。たとえば、サービスプロバイダーアクセスネットワークでは、一般的に、トラフィックフローが10ギガビットイーサネット(GE)などの高速インターフェイスからルータに入り、1 GEなどの低速インターフェイスから出る速度不一致のシナリオで発生します。

マイクロバーストがASR 920ルータで引き起こす最も一般的な問題は、出力方向でのインターフェイスパケットのドロップです。これは、非常に短い時間（ミリ秒単位）で、発信インターフェイスのレートよりも高いバーストが着信インターフェイスに発生した場合に発生します。この間、パケットはバッファされる必要があります。

ASR 920/RSP2プラットフォームでは、各1 GEインターフェイスのキューに割り当てられるデフォルトバッファは48 KBですが、各10 GEインターフェイスのキューには120 KBです。さらに、11.75 MBの共有バッファを使用できます。デフォルトインターフェイスバッファがパケットのバーストに対応するのに十分でない場合は、共有バッファを使用できます。インターフェイスの共有バッファの使用を有効にするには、そのインターフェイスでQoSポリシーを設定し、queue-limitパラメータを定義する必要があります。

キュー制限の設定

インターフェイスGE 0/0/1のqueue-limitの設定例を次に示します。

```
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
```

このポリシーはどのトラフィックとも一致しないため、このポリシーも影響を受けません。このポリシーは、インターフェイスGE 0/0/1のキューバッファを増やすだけです。

非リーフクラスでqueue-limitを設定することはサポートされていないため、コマンドmatch qos-

group 99が必要になります。このため、リーフクラスdefaultでqueue-limitコマンドを設定するには、架空のクラスマップを親クラスとして使用する必要があります。

```
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
```

キュー制限は、次のようなさまざまな方法で設定できます。

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

パラメータと値は、ネットワーク要件に従って選択する必要があります。これらのパラメータと、値の変更がバッファの使用状況に与える影響を理解するには、次のセクションで示すラボテストを参照してください。

注：ポートチャネルの物理インターフェイス部分でのみ、ポートチャネルインターフェイスでQoSポリシーを設定することはサポートされていません

```
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およびNCS 520との違い

[Cisco RSP3モジュールのQoS機能:](#)

- RSP3モジュールには、NPUごとに4 GBの外部パケットバッファがあります
- RSP3モジュールは48000キューをサポート
- デフォルトでは、RSP3モジュールはキューごとに最大1 MBのキュー制限をサポートします
- キュー制限のパーセンテージは、合計バッファの1 GBの中から考慮されます

RSP3スーパーバイザとNCS 520を搭載したルータでは、設定できるqosグループの数は0 ~ 7に制限されています。

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

NCS 520では、デフォルトですべてのインターフェイス間で2 MBのキューバッファが共有されています。キュー制限を使用したポリシーマップが設定されている場合、外部2 GBバッファにアクセスできません。また、キュー制限のbytesパラメータとusパラメータにも違いがあります。

```
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
```

キュー制限(QL)ラボテスト、バイトとパーセントの違い

テストスコープとパラメータ

前述したように、キュー制限をバイト単位で設定可能な最大値は2097152で、これはASR 920プラットフォーム(11.75 MB ~ 45898 * 256バイト)の共有バッファの約18 %です。キュー制限をパーセント単位で設定すると、最大100%まで増やすことができます。したがって、パーセントとバイトを同等の値と比較するために、テストでは117498 ~ 2097152バイトのバイト値と1 ~ 18 %のキュー制限パーセント値を使用します。

```
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
```

36のポリシーマップが設定されています。18は1 ~ 18 %の範囲のキュー制限値を持ち、他の18は117498 ~ 2097152バイトの範囲のキュー制限値を持ちます。

```
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

各ポリシーは、IXIAで生成される同じマイクロバーストトラフィックに対してテストされます。このトラフィックは、ASR 920の10 GEポートに到達し、同じルータの1 GEポートから送信されます。

バーストは、5秒間隔で4 Gbpsの20000個の packets で構成されます。これは、特定の packet サイズに対するバースト期間です。

```
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
```

テスト手順

ステップ1：発信インターフェイスでポリシーマップQUEUE-LIMIT-BYTES-X(X=1)をバイトY=117498で適用します。

ステップ2：マイクロバーストトラフィックを1分間実行します。

ステップ3：出力されるパケットの総数と、廃棄されたパケットの数を測定します。

ステップ4：ドロップされたパケットと出力パケットの総数の比率を計算する。

ステップ5：ステップ1から1回繰り返します。今回は、policy-map QUEUE-LIMIT-PERCENT-Xとqueue-limit percent Xを使用します(X=1)。

ステップ6：ポリシーマップ名とパーセント値Y=Y+117498バイトについて、ステップ1からX=X+1を繰り返します。X=18とY=2097152まで繰り返します。

例

QUEUE-LIMIT-BYTES-1 policy-mapによる測定：

```
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
```

マイクロバーストラフィックが1分間続いた後の測定：

```
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 runs, 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
```

パケットドロップデルタ : $684561562 - 684263427 = 298135$
出力デルタのパケット総数 : $7829753878 - 7829367523 = 386355$
パケットドロップとパケットアウトの比率 : $298135 / 386355 = 77\%$

ラボテストの結果

説明したように、36のポリシーマップがテストされます。1 ~ 18%の範囲のキュー制限値が設定された18個のポリシーを、117498 ~ 2097152バイトの範囲のキュー制限値が設定された他の18個のポリシーと比較してテストします。各ポリシーマップは、IXIAで生成された同じマイクロバーストトラフィックに対してテストされます。

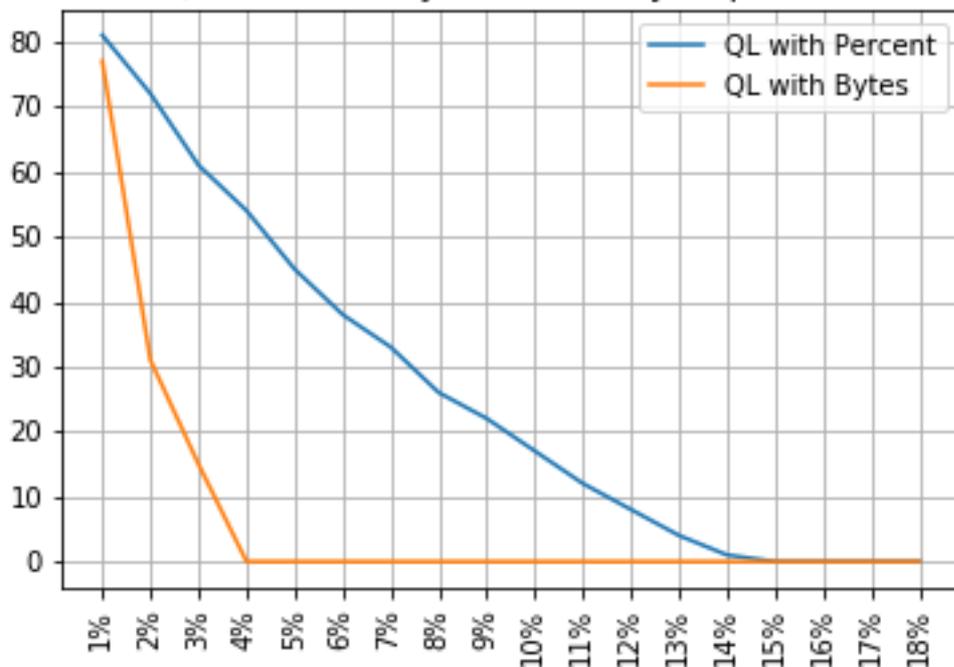
このセクションでは、異なるパケットサイズ(バイト単位)の結果を確認するために、このテストの結果を7回繰り返して公開します。64、200、300、518、800、1024、1400。

読み取りを容易にするために、パケットサイズごとに結果がテーブルに表示され、グラフが作成されます。

64バイトパケットのQLの割合とバイトの比較

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%

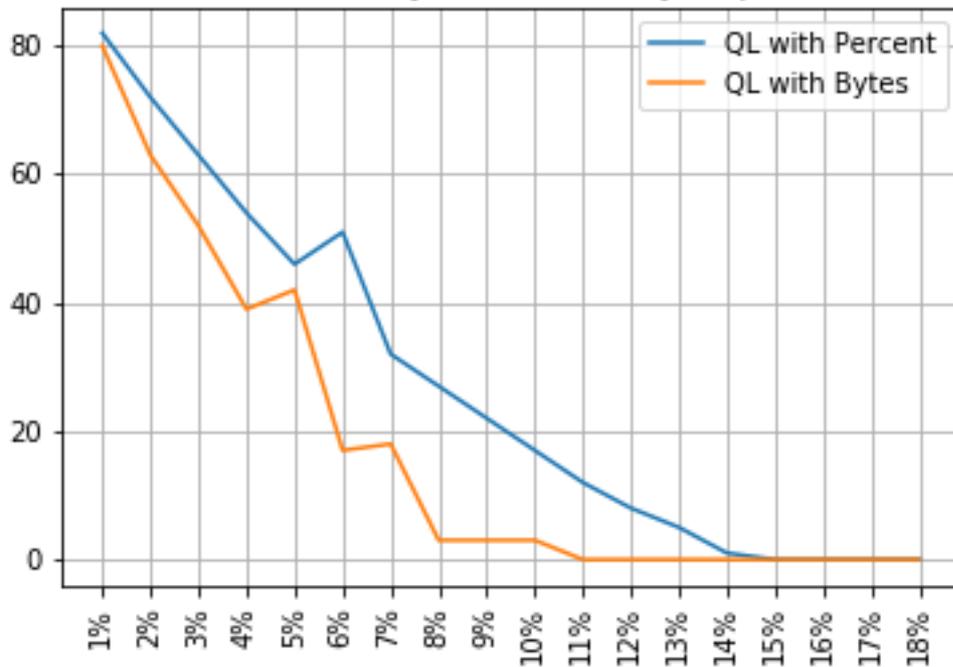
QL Percent vs bytes with 64 bytes packets



200バイトパケットのQLパーセント対バイト

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

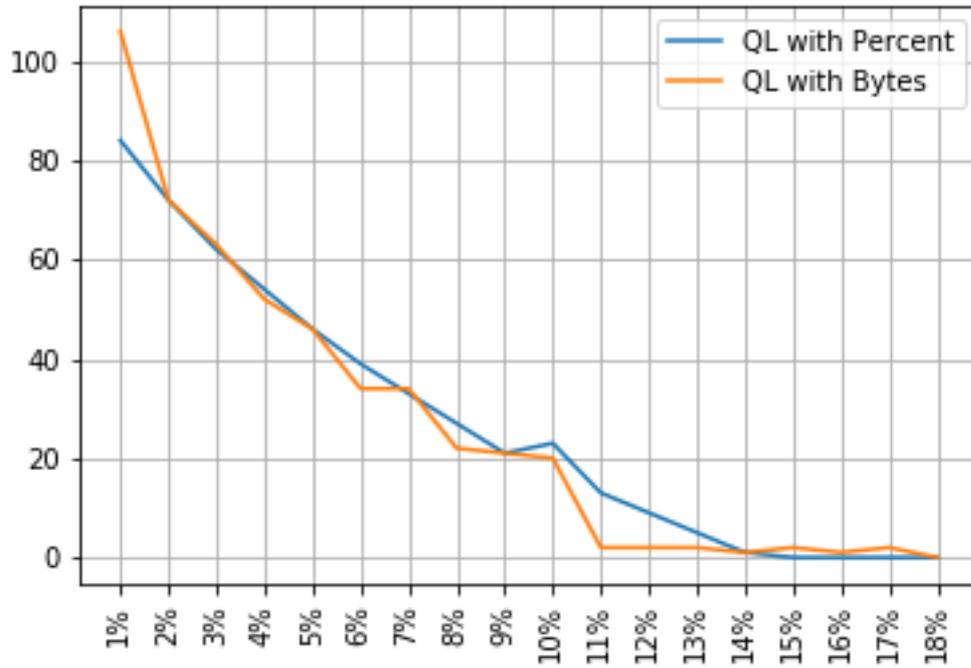
QL Percent vs bytes with 200 bytes packets



300バイトのパケットでのQLの割合とバイト

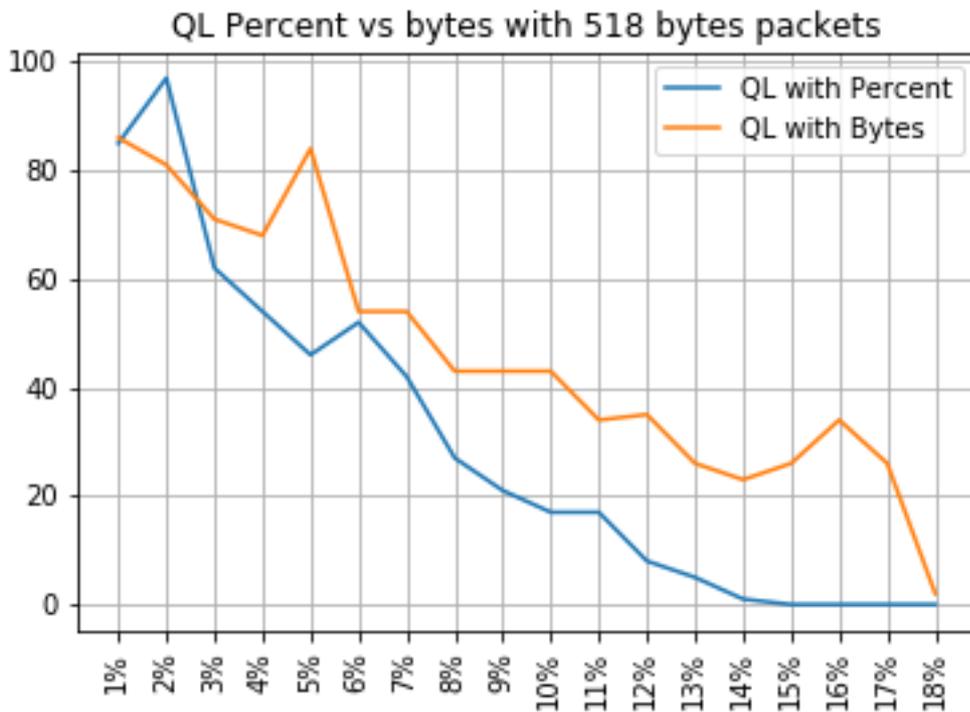
Policy-map		Drop rate with PC	Drop rate with Bytes
QL Bytes/Percent 1	1	84%	106%
QL Bytes/Percent 2	2	72%	72%
QL Bytes/Percent 3	3	62%	63%
QL Bytes/Percent 4	4	54%	52%
QL Bytes/Percent 5	5	46%	46%
QL Bytes/Percent 6	6	39%	34%
QL Bytes/Percent 7	7	33%	34%
QL Bytes/Percent 8	8	27%	22%
QL Bytes/Percent 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



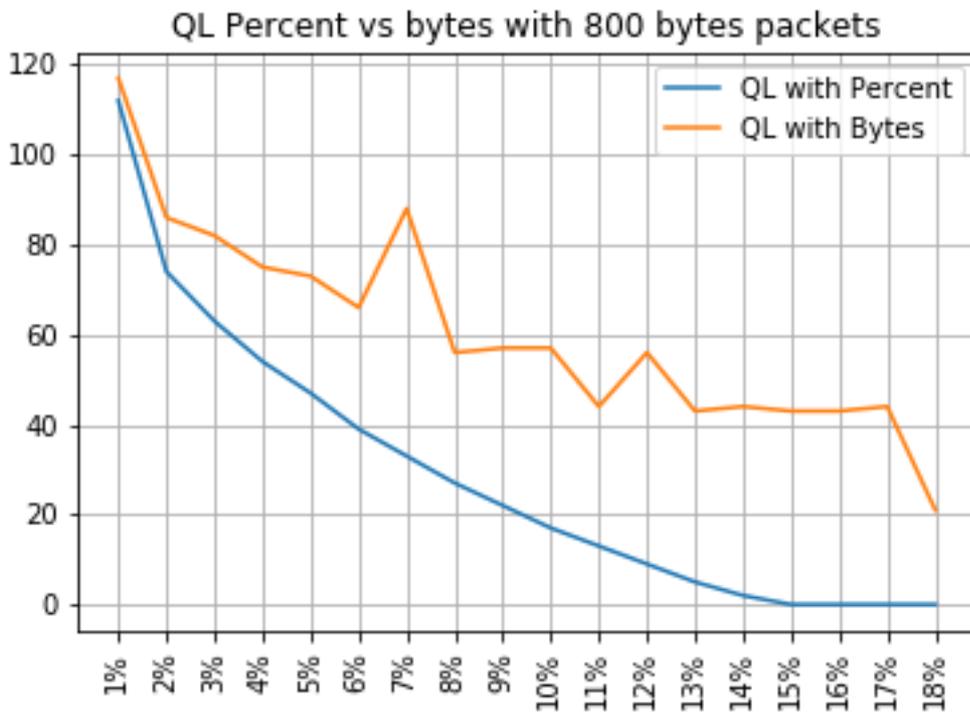
518バイトパケットのQLパーセント対バイト

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%



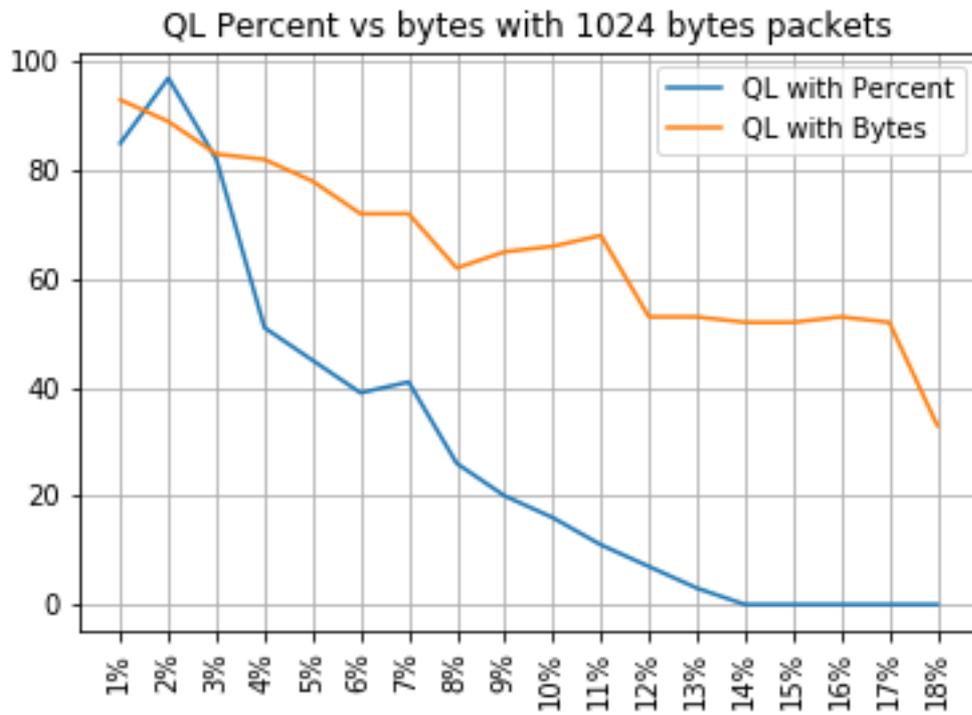
800バイトのパケットでのQLの割合とバイト

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%



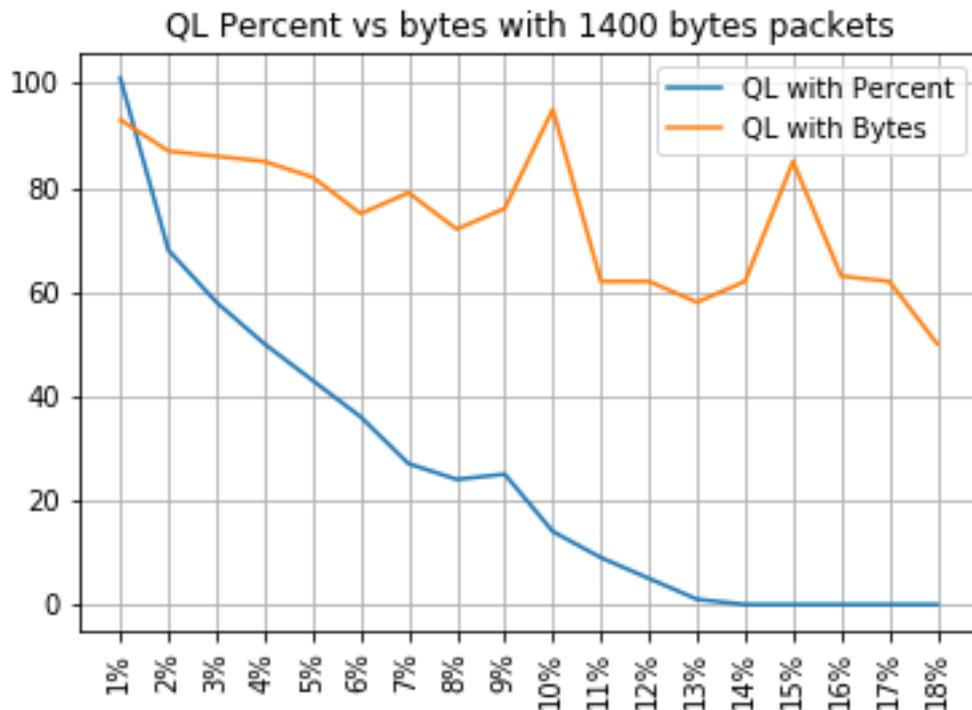
1024バイトパケットのQLパーセント対バイト

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%



1400バイトパケットのQLパーセント対バイト

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%



ドロップとパケットサイズの関係

前述したように、ASR 920には11.75 MBの内部パケットバッファがあり、それぞれ256バイトの45898 Qnodeに分割されています。

- サイズが256バイト未満のパケットの場合は、正確に1 Qnodeが使用されます
- サイズが1024バイトのパケットには、4つのQnodeが使用されます
- サイズが257バイトのパケットの場合、2つのQnodeが使用され、未使用の255バイトは失われます

したがって、小さなパケットで保存できる量よりも小さな量の大きなパケットを保存できます。同等のキュー制限サイズでのドロップレートとパケットサイズの関係が予想されます。

等価バイト値とパーセント値の差

説明したように、11.75 MBの共有バッファには45898 Qnodeがあり、計算を容易にするために45900に丸められます。

queue-limit percentの計算では、11.75 MBの割合ではなく、45900 Qnodeの割合が計算されます。したがって、queue-limit percent 10は、4590 Qnodeを提供する45900の10%を意味します。さらに、割り当てられたQnodeの割合は、サイズとは無関係に、キューに格納できるパケットの数と見なされます。前の例に戻ると、次のようになります。

queue-limit percent 10 = 4590 Qnodes = 4590パケット。

この計算はパケットサイズに依存しないため、サイズが256バイト以下のパケットでは、実際には1つのQnodeのみが使用され、Qnodeとパケットの間の等価性は維持されます。

queue-limit percent 10 = 4590 Qnodes = 4590パケットの256バイト = 4590*256バイト = 1.175 MB = バッファの10%

ただし、大きなパケットの場合は、バッファのより大きな部分が割り当てられます。たとえば、次に1024バイトのパケットの計算を示します。この計算では、各パケットが4つのQnodeを消費します。

$\text{queue-limit percent } 10 = 4590 \text{ Qnodes} = 1024\text{バイトの}4590\text{パケット} = 4590 * 4 * 256\text{バイト} = 4.7 \text{ MB} = \text{バッファの}40\%$

注意： queue-limit percentの値を高く設定することは推奨されません。

queue-limit percentの値を高く設定すると、1つのインターフェイスが一時的に11.75 MBの共有バッファすべてを占有する可能性があります。

まとめ

- 小さなパケットではqueue-limitバイトの効率が良いことがわかります。queue-limit bytes <x>はqueue-limit <x>よりも優れた動作を行い、最大300バイトまで処理できます
- 300バイトのパケットでは、queue-limit bytesとqueue-limit percent efficiencyは同じです
- 300バイトを超えるパケットサイズでは、キュー制限の割合がより効率的です。インターネットトラフィックは平均518バイトであるため、お客様から報告されたキュー制限の割合は、実際のシナリオでより多くの利点があることを意味します
- キュー制限の割合の効率は、パケットサイズに比例して向上します (パケットが大きいほど、キュー制限の割合はキュー制限バイトよりも効率的です)
- 256 MBを超えるサイズのパケットに対するバッファ領域の割り当てにより大きな値を与えるために、キュー制限パーセントが実装されています

実際のトラフィックシナリオにおけるパーセント値の近似方法

サイズが256バイトでキュー制限が10%のパケットがある場合は、次の等価性が有効であることがわかっています。

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

512バイトのパケットの場合は使用率が2倍になり、1024バイトのパケットの場合は4倍になります。

つまり、実際のキュー制限はバッファの少なくとも10%で、最大MTUを1500バイトと仮定した場合、1つのパケットを格納するには6つのQnodeが必要です。これにより、最大キュー制限は次のようになります。

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

この方法では、バッファ使用率の下限と上限をqueue-limit percent 10で定義できます。そのため、一般的に言えば、バッファ使用率の平均は次のようになります。

$\text{ceil}(\text{avg_pkt_size}/256) * ((\text{qlimit_percent}/45900) * 100)$

ラボ機器の例：

```
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_sizeは、8870166880/11247114 ~ 788バイトで計算できます。
```

queue-limit percent 10を使用すると、バッファの平均最大使用率は次のようになります。

```
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
```

マイクロバーストバッファの使用状況を確認する方法

16.9.3よりも前のCisco IOS XEリリースでは、インターフェイスの共有バッファは、データパケットと制御パケット (BFD、ルーティングプロトコル、ARP、LDP、パントキープアライブなど) の両方に使用されていました。 インスタントバッファの使用状況を確認するには、次のコマンドを使用できます。

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

16.9.3以降では、バッファの使用率を改善するためにいくつかの変更が導入され、2つに分割されました。1024エントリ(256 KB)は制御トラフィック用に予約され、残りはデータトラフィック用に予約されています。

この場合のバッファの使用状況は、次のコマンドで監視できます。

```
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)
```

バッファがマイクロバーストを処理するため、コマンドを何度も繰り返して、reservedUsedCountの値が0以外であることを確認する必要があります。

バッファの使用量はreservedUsedCount/reservedFreeCountで簡単に計算できます。たとえば、 $8114/37784 = 21,5\%$ を使用します。バーストが終了すると、バッファはすぐに0に戻るか、またはバッファの近くに戻ります。

Cisco IOS XEリリース17.6.1以降では、データトラフィックと制御トラフィックの両方にバッファ全体を使用するか（ネットワークで制御トラフィックのレートが高い場合に推奨）、前述のようにバッファを2に分割するかを選択できます。次の手順の設定に従って選択します（デフォルトでは無効）。

```
ACDC-920-1(config)#platform qos-buffer enhance enable
```

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

Cisco IOS XEリリース17.7.1以降では、制御トラフィックに割り当てるサイズを選択することもできます。

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

場所：

- 1は256 KBの制御バッファを示します。
- 2は500 KB
- 3は756 KB
- 4は1 MBを示します。

翻訳について

シスコは世界中のユーザにそれぞれの言語でサポート コンテンツを提供するために、機械と人による翻訳を組み合わせて、本ドキュメントを翻訳しています。ただし、最高度の機械翻訳であっても、専門家による翻訳のような正確性は確保されません。シスコは、これら翻訳の正確性について法的責任を負いません。原典である英語版（リンクからアクセス可能）もあわせて参照することを推奨します。