

Gestion des microrafales dans ASR 920, différences entre les octets et le pourcentage avec Queue-Limit

Contenu

[Introduction](#)

[Conditions préalables](#)

[Conditions requises](#)

[Components Used](#)

[Informations générales](#)

[Configurer Queue-Limit](#)

[Différences avec RSP3 et NCS 520](#)

[Test de TP QL \(Queue-Limit\), différences entre les octets et le pourcentage](#)

[Portée et paramètres du test](#)

[Procédure D'Essai](#)

[Exemple](#)

[Résultats des tests en laboratoire](#)

[Pourcentage QL par rapport aux octets avec des paquets de 64 octets](#)

[Pourcentage QL par rapport aux octets avec des paquets de 200 octets](#)

[Pourcentage QL par rapport aux octets avec des paquets de 300 octets](#)

[Pourcentage QL vs octets avec 518 octets paquets](#)

[Pourcentage QL par rapport aux octets avec 800 octets paquets](#)

[Pourcentage QL par rapport aux octets avec des paquets de 1024 octets](#)

[Pourcentage QL par rapport aux octets avec des paquets de 1 400 octets](#)

[Relation entre les abandons et la taille des paquets](#)

[Différence entre les octets équivalents et les valeurs en pourcentage](#)

[Conclusions](#)

[Comment approximer les valeurs en pourcentage dans les scénarios de trafic réel](#)

[Vérification de l'utilisation de la mémoire tampon Microburst](#)

Introduction

Ce document décrit comment gérer les microrafales dans les routeurs ASR 920, qui sont souvent la cause des pertes de paquets de sortie d'interface. Il montre en détail la différence entre l'utilisation des octets et le pourcentage avec la commande queue-limit.

Conditions préalables

Conditions requises

Cisco vous recommande de prendre connaissance des rubriques suivantes :

Components Used

Les informations de ce document sont basées sur un routeur ASR 920 qui exécute la version logicielle Cisco IOS-XE 16.9.6.

IXIA est utilisé comme générateur de trafic pour les tests de laboratoire.

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. Si votre réseau est en ligne, assurez-vous de bien comprendre l'incidence possible des commandes.

Informations générales

Les microrafales sont appelées petites pointes dans le trafic réseau. Dans les réseaux Service Provider Access, par exemple, cela se produit généralement dans des scénarios de non-concordance de vitesse où un flux de trafic entre dans le routeur à partir d'une interface haut débit, comme 10 Gigabit Ethernet (GE), et sort par une interface bas débit, comme 1 GE.

Le problème le plus courant causé par les microbutées dans les routeurs ASR 920 est les pertes de paquets d'interface dans la direction de sortie. Cela se produit lorsqu'il y a des rafales sur l'interface entrante qui sont supérieures au débit de l'interface sortante, pendant une très courte période de temps (de l'ordre de millisecondes). Pendant ce temps, les paquets doivent être mis en mémoire tampon.

Sur les plates-formes ASR 920 / RSP2, la mémoire tampon par défaut allouée aux files d'attente sur chaque interface 1 GE est de 48 Ko, tandis que pour les files d'attente sur chaque interface 10 GE est de 120 Ko.

En plus de cela, il y a une mémoire tampon partagée disponible de 11,75 Mo. Lorsque le tampon d'interface par défaut n'est pas suffisant pour accueillir une rafale de paquets, le tampon partagé peut être utilisé.

Afin d'activer l'utilisation de la mémoire tampon partagée pour une interface, il est nécessaire de configurer une stratégie QoS sous cette interface, qui définit le paramètre queue-limit.

Configurer Queue-Limit

Voici un exemple de configuration de queue-limit pour l'interface 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
```

Cette stratégie ne correspond à aucun trafic. Par conséquent, elle ne l'affecte pas non plus. Tout ce que fait cette politique est d'augmenter le tampon de file d'attente de l'interface GE 0/0/1.

La commande **match qos-group 99** est nécessaire parce qu'elle n'est pas prise en charge pour configurer queue-limit dans une classe non-leaf, donc vous devez utiliser une class-map fictive comme classe parent afin de configurer la commande queue-limit sous la classe leaf-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
```

La queue-limit peut être définie de différentes manières :

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

Le paramètre et la valeur doivent être sélectionnés en fonction des exigences de votre réseau. Afin de comprendre ces paramètres et comment une modification des valeurs affecte l'utilisation de la mémoire tampon, vous pouvez vous reporter au test de TP présenté dans la section suivante.

Note: Il n'est pas possible de configurer une stratégie QoS sous des interfaces port-channel, mais uniquement sous la partie interfaces physiques du port-channel

```
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
```

Différences avec RSP3 et NCS 520

[Fonctionnalités QoS du module Cisco RSP3](#) :

- Le module RSP3 dispose de tampons de paquets externes de 4 Go par NPU
- Le module RSP3 prend en charge 48000 files d'attente
- Par défaut, le module RSP3 prend en charge jusqu'à 1 Mo de file d'attente par file d'attente
- Le pourcentage de limite de file d'attente est considéré sur 1 Go du total des tampons

Pour les routeurs avec superviseurs RSP3 et NCS 520, le nombre de groupes QoS pouvant être configurés est limité à 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
```

Dans le NCS 520, il y a une mémoire tampon de file d'attente de 2 Mo partagée entre toutes les interfaces par défaut, une mémoire tampon externe de 2 Go est accessible quand un policy-map avec queue-limit est configuré. Il existe également une différence dans les paramètres bytes et use pour la 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 de TP QL (Queue-Limit), différences entre les octets et le pourcentage

Portée et paramètres du test

Comme indiqué précédemment, la valeur maximale configurable pour queue-limit en octets est 2097152, ce qui représente environ 18 % de la mémoire tampon partagée sur les plates-formes ASR 920 (11,75 Mo ~ 45898 * 256 octets).

Si vous configurez la limite de file d'attente en pourcentage, vous pouvez aller jusqu'à 100 %. Par conséquent, afin de comparer le pourcentage et les octets avec des valeurs équivalentes, le test prend des valeurs d'octets de 117498 à 2097152 octets et des valeurs de pourcentage de limite de file d'attente de 1 % à 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
```

36 cartes-politiques sont configurées : 18 avec des valeurs queue-limit comprises entre 1 % et 18 % et les 18 autres avec des valeurs queue-limit comprises entre 117498 et 2097152 octets.

```
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

Chaque stratégie est testée par rapport au même trafic de microrafales, généré avec IXIA. Ce trafic arrive sur un port 10 GE du routeur ASR 920 et sort sur un port 1 GE du même routeur.

Les rafales se composent de 20000 paquets à 4 Gbits/s à intervalles de 5 secondes. Il s'agit de la durée de rafale donnée pour une taille de paquet spécifique :

```
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
```

Procédure D'Essai

Étape 1 : application de la carte de stratégie QUEUE-LIMIT-BYTES-X (X=1) avec les octets Y=117498 sous l'interface de sortie

Étape 2 : exécution du trafic en microrafale pendant 1 minute

Étape 3. Mesurez le nombre total de paquets en sortie et le nombre de paquets abandonnés.

Étape 4 : calcul du rapport entre les paquets abandonnés et le nombre total de paquets en sortie

Étape 5. Répétez une fois à partir de l'étape 1, cette fois utilisez policy-map QUEUE-LIMIT-PERCENT-X avec queue-limit percent X, où X=1.

Étape 6. Répétez l'étape 1 avec X=X+1 pour le nom de la carte de stratégie et la valeur de pourcentage, Y=Y+117498 octets. Répéter jusqu'à X=18 et Y=2097152.

Exemple

Mesure avec la carte-politique 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

Mesure après 1 minute de trafic en microrafale :

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

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 des pertes de paquets : $684561562 - 684263427 = 298135$

Delta total de sortie des paquets : $7829753878 - 7829367523 = 386355$

Rapport entre les abandons de paquets et les paquets sortants : $298135 / 386355 = 77 \%$

Résultats des tests en laboratoire

Comme expliqué, 36 cartes-politiques sont testées : 18 configurées avec des valeurs de limite de file d'attente comprises entre 1 % et 18 % sont testées par rapport aux 18 autres stratégies configurées avec des valeurs de limite de file d'attente comprises entre 117498 et 2097152 octets. Chaque policy-map est testée par rapport au même trafic de microrafales, généré avec IXIA.

Dans cette section sont exposés les résultats de ce test, répétés 7 fois afin de vérifier les résultats avec différentes tailles de paquets, en octets : 64, 200, 300, 518, 800, 1024, 1400.

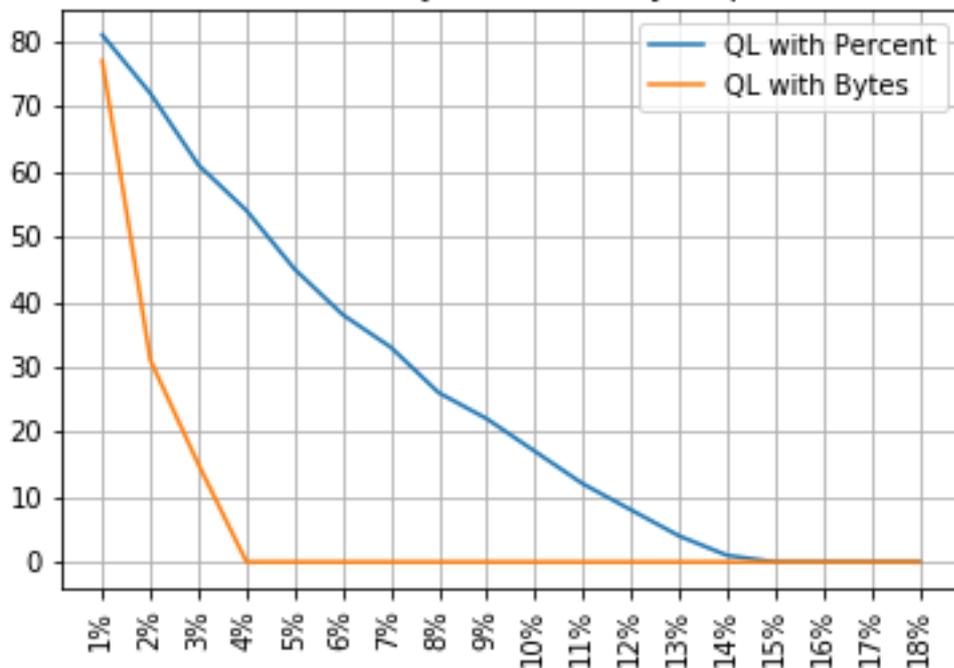
Afin de faciliter la lecture, pour chaque taille de paquet, les résultats sont exposés dans une table puis représentés graphiquement.

Pourcentage QL par rapport aux octets avec des paquets de 64 octets

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%

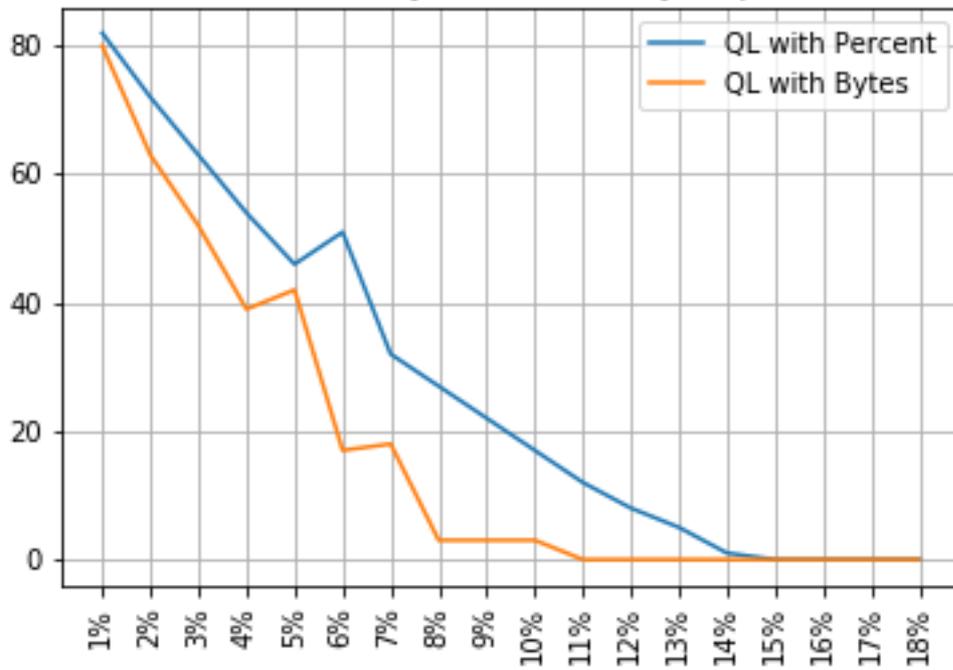
QL Percent vs bytes with 64 bytes packets



Pourcentage QL par rapport aux octets avec des paquets de 200 octets

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%

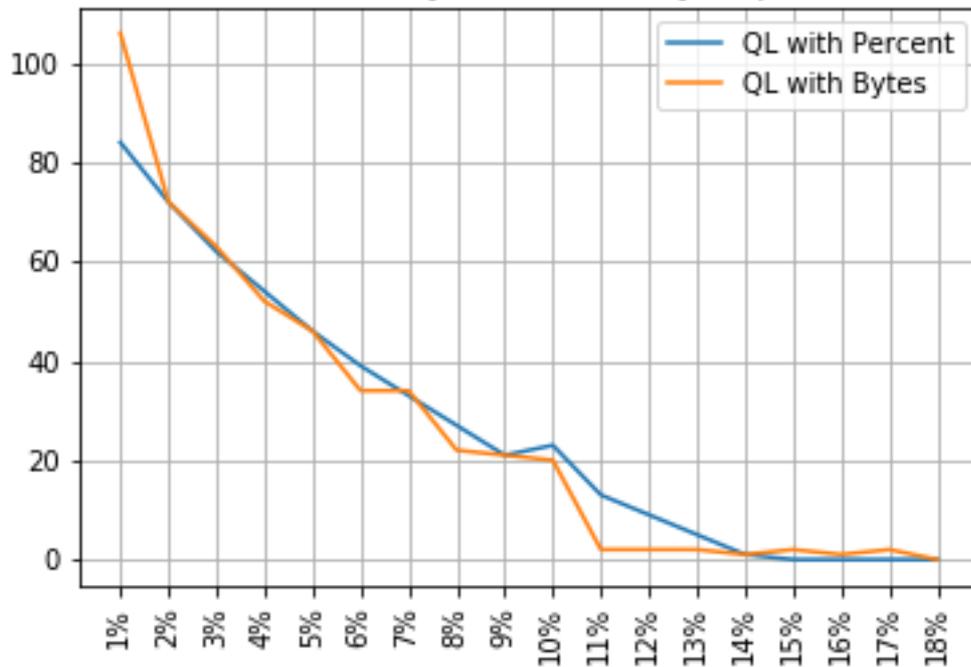
QL Percent vs bytes with 200 bytes packets



Pourcentage QL par rapport aux octets avec des paquets de 300 octets

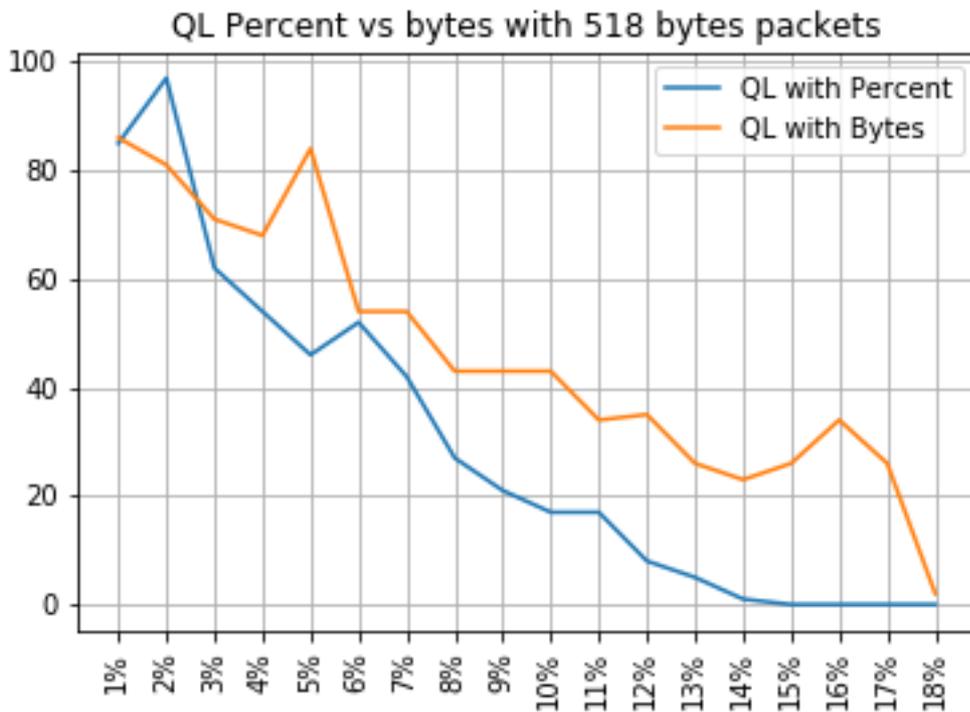
Policy-map		Drop rate with PC	Drop rate with Bytes
QL Bytes/Percent 1	1	84%	106%
QL Bytes/Percent 2	2	72%	72%
QL Bytes/Percent 3	3	62%	63%
QL Bytes/Percent 4	4	54%	52%
QL Bytes/Percent 5	5	46%	46%
QL Bytes/Percent 6	6	39%	34%
QL Bytes/Percent 7	7	33%	34%
QL Bytes/Percent 8	8	27%	22%
QL Bytes/Percent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



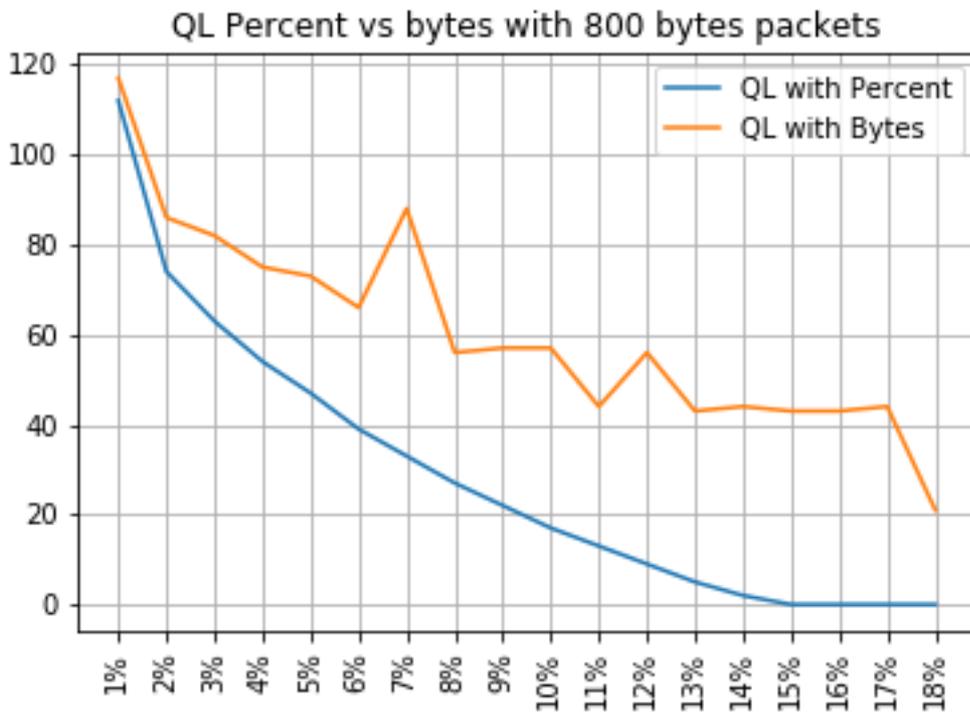
Pourcentage QL vs octets avec 518 octets paquets

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



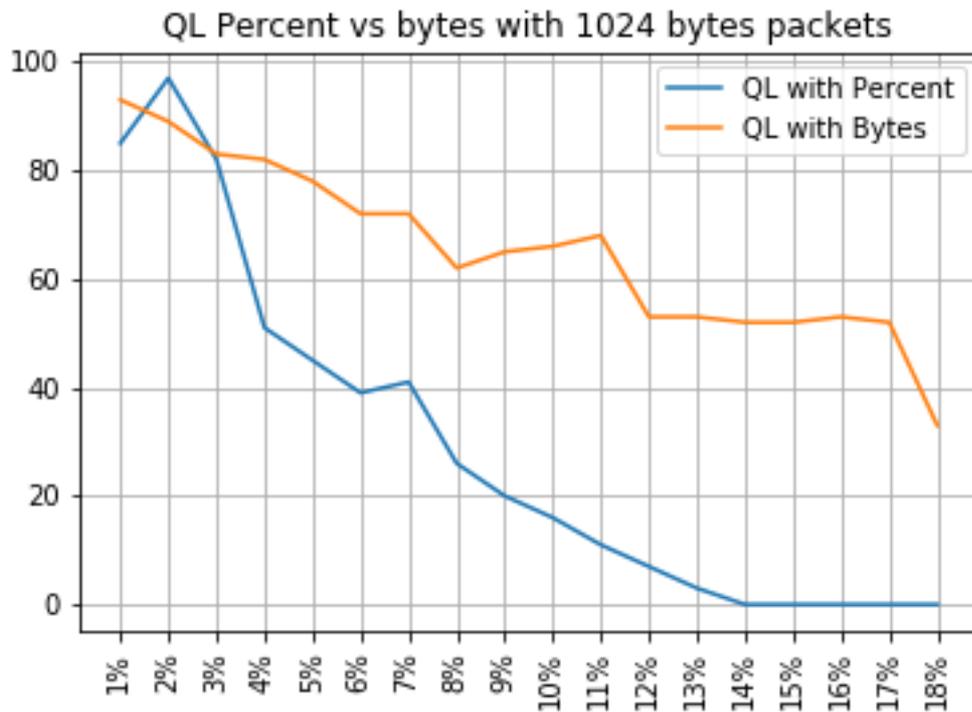
Pourcentage QL par rapport aux octets avec 800 octets paquets

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%



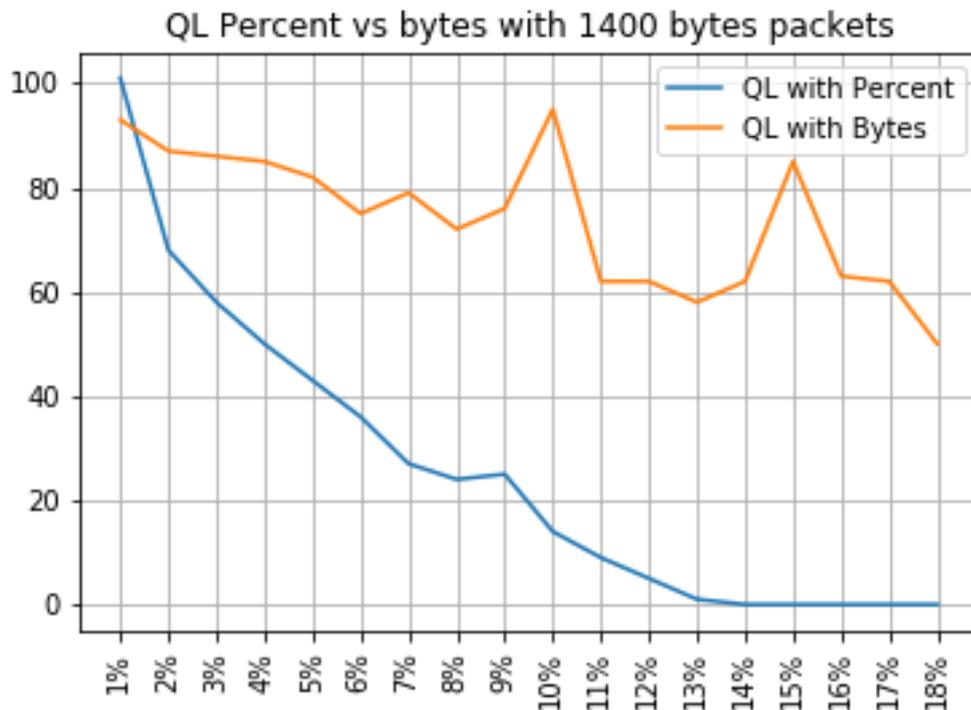
Pourcentage QL par rapport aux octets avec des paquets de 1024 octets

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%



Pourcentage QL par rapport aux octets avec des paquets de 1 400 octets

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%



Relation entre les abandons et la taille des paquets

Comme mentionné, l'ASR 920 a un tampon de paquets interne de 11,75 Mo qui est divisé en 45898 Qnoeuds de 256 octets chacun.

- Pour un paquet dont la taille est inférieure à 256 octets, exactement 1 Qnode est utilisé
- Pour un paquet de taille 1024 octets, 4 Qnodes sont utilisés
- Pour un paquet de taille 257 octets, 2 Qnodes sont utilisés et les 255 octets inutilisés sont perdus

Par conséquent, vous pouvez stocker une plus petite quantité de grands paquets que ce que vous pouvez stocker avec de petits paquets. La relation entre le taux d'abandon et la taille des paquets à des tailles de file d'attente limitées équivalentes est attendue.

Différence entre les octets équivalents et les valeurs en pourcentage

Comme expliqué, il y a 45898 Qnodes dans la mémoire tampon partagée de 11,75 Mo, arrondie à 45900 pour faciliter le calcul.

Le calcul du pourcentage de limite de file d'attente ne calcule pas le pourcentage des 11,75 Mo mais celui des 45900 Qnodes. Ainsi, le pourcentage de limite de file d'attente 10 signifie 10 % de 45900, ce qui donne 4590 Qnodes.

En outre, le pourcentage de Qnodes alloué est considéré comme le nombre de paquets qui peuvent être stockés dans la file d'attente, indépendamment de sa taille. Pour revenir à l'exemple précédent, cela signifie que :

queue-limit percent 10 = 4590 Qnodes = 4590 paquets.

Comme ce calcul est indépendant de la taille du paquet, pour les paquets de 256 octets ou moins, un seul Qnode est réellement utilisé et l'équivalence entre Qnodes et les paquets est maintenue :

queue-limit percent 10 = 4590 Qnodes = 4590 paquets de 256 octets = 4590*256 octets = 1,175 Mo = 10 % de la mémoire tampon

Cependant, avec des paquets plus importants, une partie plus généreuse de la mémoire tampon est allouée. Par exemple, il s'agit du calcul pour les paquets de 1024 octets, où chaque paquet consomme 4 Qnodes :

queue-limit percent 10 = 4 590 Qnodes = 4 590 paquets de 1 024 octets = 4 590*4*256 octets = 4,7 Mo = 40 % de la mémoire tampon

Attention : il n'est pas recommandé de configurer des valeurs élevées de pourcentage de limite de file d'attente.

Si vous configurez des valeurs élevées de pourcentage de limite de file d'attente, une seule interface peut occuper temporairement toute la mémoire tampon partagée de 11,75 Mo.

Conclusions

- Vous pouvez clairement voir que l'efficacité des octets de queue-limit est meilleure avec les petits paquets - les octets de queue-limit <x> fonctionnent mieux que queue-limit <x> pour un maximum de 300 octets
- Avec un paquet de 300 octets, les octets de limite de file d'attente et le pourcentage d'efficacité de limite de file d'attente sont identiques
- Taille de paquet supérieure à 300 octets, le pourcentage de limite de file d'attente est plus efficace. Comme le trafic Internet est en moyenne de 518 octets, cela signifie que les scénarios de la vie réelle bénéficient davantage du pourcentage de limite de file d'attente, comme indiqué par les clients
- L'efficacité du pourcentage de limite de file d'attente s'améliore linéairement par rapport à la taille du paquet (plus les paquets sont gros, plus le pourcentage de limite de file d'attente est efficace par rapport aux octets de limite de file d'attente)
- Le pourcentage de limite de file d'attente est mis en oeuvre pour être plus généreux dans l'allocation d'espace tampon pour les paquets de plus de 256 Mo

Comment approximer les valeurs en pourcentage dans les scénarios de trafic réel

Dans le cas où vous avez des paquets avec une taille de 256 octets et une limite de file d'attente de 10%, vous savez déjà que cette équivalence est valide :

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

Avec des paquets de 512 octets seulement, l'utilisation est le double, avec 1024 octets seulement des paquets, il est quatre fois plus, et ainsi de suite.

Cela signifie que la limite de file d'attente réelle est d'au moins 10% de la mémoire tampon et, si vous supposez une MTU maximale de 1500 octets, vous avez besoin de 6 Qnodes pour stocker un seul paquet, ce qui donne une limite de file d'attente maximale de :

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

De cette façon, vous pouvez définir les limites inférieure et supérieure de l'utilisation de la mémoire tampon avec le pourcentage de limite de file d'attente 10, donc plus généralement l'utilisation maximale moyenne de la mémoire tampon est approximativement :

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

Exemple d'un équipement de TP :

```
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
```

Vous pouvez calculer la taille_pkt_moyenne en $8870166880/11247114 \sim 788$ octets.

Avec un pourcentage limite de file d'attente de 10, vous avez une utilisation maximale moyenne de la mémoire tampon de :

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

Calculaton 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
```

Vérification de l'utilisation de la mémoire tampon Microburst

Dans les versions de Cisco IOS-XE antérieures à la version 16.9.3, la mémoire tampon partagée des interfaces était utilisée pour les paquets de données et de contrôle (tels que BFD, les protocoles de routage, ARP, LDP, les keepalives punt). Afin de vérifier l'utilisation instantanée de

la mémoire tampon, vous pouvez utiliser cette commande :

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

Après la version 16.9.3, certaines modifications ont été apportées pour améliorer l'utilisation de la mémoire tampon et elle a été divisée en 2 : 1 024 entrées (256 Ko) ont été réservées au trafic de contrôle et le reste est réservé au trafic de données.

L'utilisation de la mémoire tampon dans ce cas peut être surveillée avec ces commandes :

```
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)
```

Notez que, étant donné que la mémoire tampon gère les microrafales, vous devez répéter la commande plusieurs fois pour voir la valeur reservedUsedCount différente de 0.

L'utilisation de la mémoire tampon peut simplement être calculée avec $\text{reservedUsedCount}/\text{reservedFreeCount}$, par exemple $8114/37784 = 21,5\%$ utilisé. Une fois la rafale terminée, la mémoire tampon doit rapidement revenir à 0 ou à proximité de.

À partir de la version 17.6.1 de Cisco IOS-XE, il est possible d'utiliser la totalité du tampon pour le trafic de données et de contrôle (de préférence si votre réseau a un débit élevé de trafic de contrôle) ou de diviser le tampon en 2 comme décrit précédemment. Le choix se fait avec la configuration de cette instruction (désactivée par défaut) :

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

À partir de Cisco IOS-XE version 17.7.1, il est également possible de choisir la taille à allouer au trafic de contrôle :

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

Where:

- 1 indique une mémoire tampon de contrôle de 256 Ko
- 2 indique 500 Ko
- 3 indique 756 Ko
- 4 indique 1 Mo

À propos de cette traduction

Cisco a traduit ce document en traduction automatisée vérifiée par une personne dans le cadre d'un service mondial permettant à nos utilisateurs d'obtenir le contenu d'assistance dans leur propre langue.

Il convient cependant de noter que même la meilleure traduction automatisée ne sera pas aussi précise que celle fournie par un traducteur professionnel.