

# WRED et MDRR sur le routeur Internet de la gamme Cisco 12000 avec un exemple de mélange de configuration Unicast, Multicast et trafic vocal

## Contenu

[Introduction](#)

[Conditions préalables](#)

[Conditions requises](#)

[Composants utilisés](#)

[Conventions](#)

[Informations générales](#)

[Classes de priorité utilisées dans le test](#)

[Coloration des paquets IP](#)

[Résultats prévus](#)

[Configuration du réseau](#)

[implémentation de Mise en file d'attente d'Engine 2 du GbE 3-Port](#)

[Algorithme WRED d'Engine 2 en logiciel de Cisco IOS](#)

[Configuration QoS](#)

[Cos de Rx](#)

[Tx QoS](#)

[Mappage d'interface](#)

[Commencer sans des baisses](#)

[Quatre flux de données de 151Mb chacun](#)

[Quatre flux de données de 160Mb chacun](#)

[Quatre flux de données de 167Mb chacun](#)

[Quatre flux de données de 191Mb chacun](#)

[Quatre flux de données de 244Mb chacun](#)

[Tout le QoS a retiré](#)

[Quatre flux de données de 153Mb chacun](#)

[Quatre flux de données de 158Mb chacun](#)

[Ajouter le chargement d'entrée](#)

[12-RP#show relie g 6/0](#)

[Changer la taille des flots](#)

[Vérifier avec un linecard de l'engine 4 des Gigabit Ethernet 10-Port](#)

[Informations connexes](#)

## [Introduction](#)

Ce document explique comment configurer un linecard de gamme Cisco 12000 pour le Détection précoce directe pondérée (WRED), décrit dans [RFC 2309](#) , dans un environnement interarmées.

## Conditions préalables

### Conditions requises

Les lecteurs de ce document doivent avoir une bonne connaissance de ce qui suit :

- [Compréhension et configuration de MDRR et de WRED sur le Routeur Internet de la série Cisco 12000](#)
- [Comment lire la sortie de la commande show controller frfab | Commandes tofab queue sur un routeur Internet de la gamme Cisco 12000](#)
- [Type de service dans l'ensemble des protocoles IP \(Internet Protocol\), priorité \(RFC-1349\)](#)
- [Dépistage précoce aléatoire pesé](#)

### Composants utilisés

Les informations dans ce document sont basées sur le logiciel et les versions de matériel ci-dessous :

- Toute version logicielle de Cisco IOS® qui prend en charge le Routeur Internet de la série Cisco 12000. Habituellement ce sont les 12.0S et les 12.0ST releases.
- Toutes les Plateformes de Cisco 12000 sont couvertes par ce document. Ceux-ci incluent les 12008, 12012, 12016, 12404, 12406, 12410, et les 12416.

Les informations contenues dans ce document ont été créées à partir des périphériques d'un environnement de laboratoire spécifique. Tous les périphériques utilisés dans ce document ont démarré avec une configuration effacée (par défaut). Si votre réseau est opérationnel, assurez-vous que vous comprenez l'effet potentiel de toute commande.

### Conventions

Pour plus d'informations sur les conventions des documents, référez-vous aux [Conventions utilisées pour les conseils techniques de Cisco](#).

## Informations générales

La gamme Cisco 12000 est l'une des Plateformes les plus populaires utilisées pour établir un réseau de noyau IP de bande passante élevée. Cette plate-forme offre la possibilité exclusive pour configurer le Qualité de service (QoS).

Puisqu'il est de plus en plus commun pour mélanger différents types de trafic IP (tel que la voix sur ip - VoIP - et Multidiffusion) dans le même réseau, les conditions requises pour la hiérarchisation et un comportement chutant commandé deviennent extrêmement importantes, et, dans de nombreux cas, sont la différence entre le succès et échec en lançant un nouveau service tel que le VoIP.

Les spécifications du réseau pour différents types de trafic IP sont hors de portée de ce document.

Ce document se concentre sur des essais en laboratoire effectués pour trouver une configuration qui s'applique sur différents linecards, y compris la gamme Cisco 12000, linecard des Gigabit Ethernet 3-Port (GbE 3-Port). Les résultats de ces tests prouvent que le linecard d'Engine 2 du GbE 3-Port est bien adapté pour un environnement de réseau impliquant un mélange de Voix, de données, et de trafic de multidiffusion. Il s'avère également que cela configurer QoS fait à une véritable différence dans un réseau congestionné.

## Classes de priorité utilisées dans le test

Les valeurs de priorité qui sont assignées aux classes différentes doivent être identiques dans le tout le réseau. Vous devez déterminer une stratégie générique.

Classe	Priorité	Le trafic
Le trafic mauvais		Tout le non identifié outre du trafic net (hors fonction-net)
Sur le net ----- sur le net	1	Trafiquez que séjours dans le réseau de fournisseur de services (sur le réseau)
Services de fournisseur de services Internet (ISP)	2	Le trafic ISP, SMTP, BRUIT, FTP, DN, telnet, SSH, WWW, https
SME (petite et moyenne entreprise)	3	Clients de l'entreprise, un service d'or
Temps réel, non vocal	4	TV, jeu en temps réel
Voix	5	Le trafic VoIP de RTP
Messages de contrôle de réseau	6-7	Protocole BGP (Border Gateway Protocol) et d'autres messages de contrôle

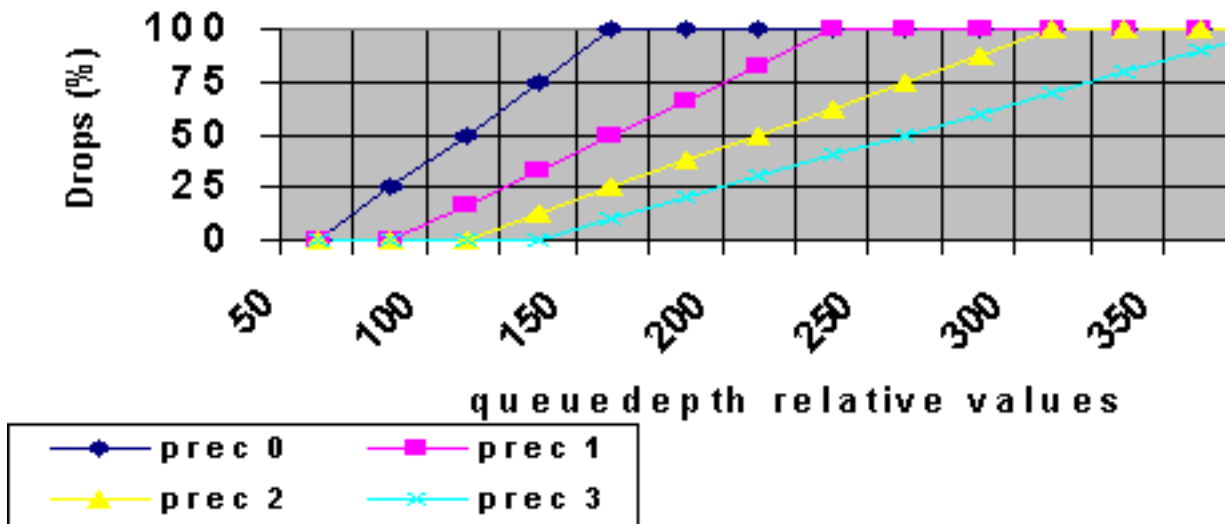
## Coloration des paquets IP

Si QoS doit être mis en application au centre d'un réseau, une condition préalable est que le fournisseur de services soit dans le plein contrôle de la valeur de priorité de *tous les* paquets IP transportés dans le réseau. La seule manière de faire ceci est de marquer tous les paquets quand ils entrent dans le réseau, ne faisant aucune distinction s'ils proviennent le côté de client/utilisateur final ou de l'Internet. Aucun marquage ou coloration ne devrait être fait au centre.

## Résultats prévus

Le but avec cette conception est d'avoir le véritable comportement WRED dans les classes 0-3. Ceci signifie que nous voudrions avoir une situation où nous commençons la priorité chutante 0 paquets pendant l'encombrement. Après cela, nous devrions également commencer à relâcher la priorité 1 si l'encombrement continue, et puis également la priorité 2 et 3. Ceci est tout décrit dans le graphique ci-dessous.

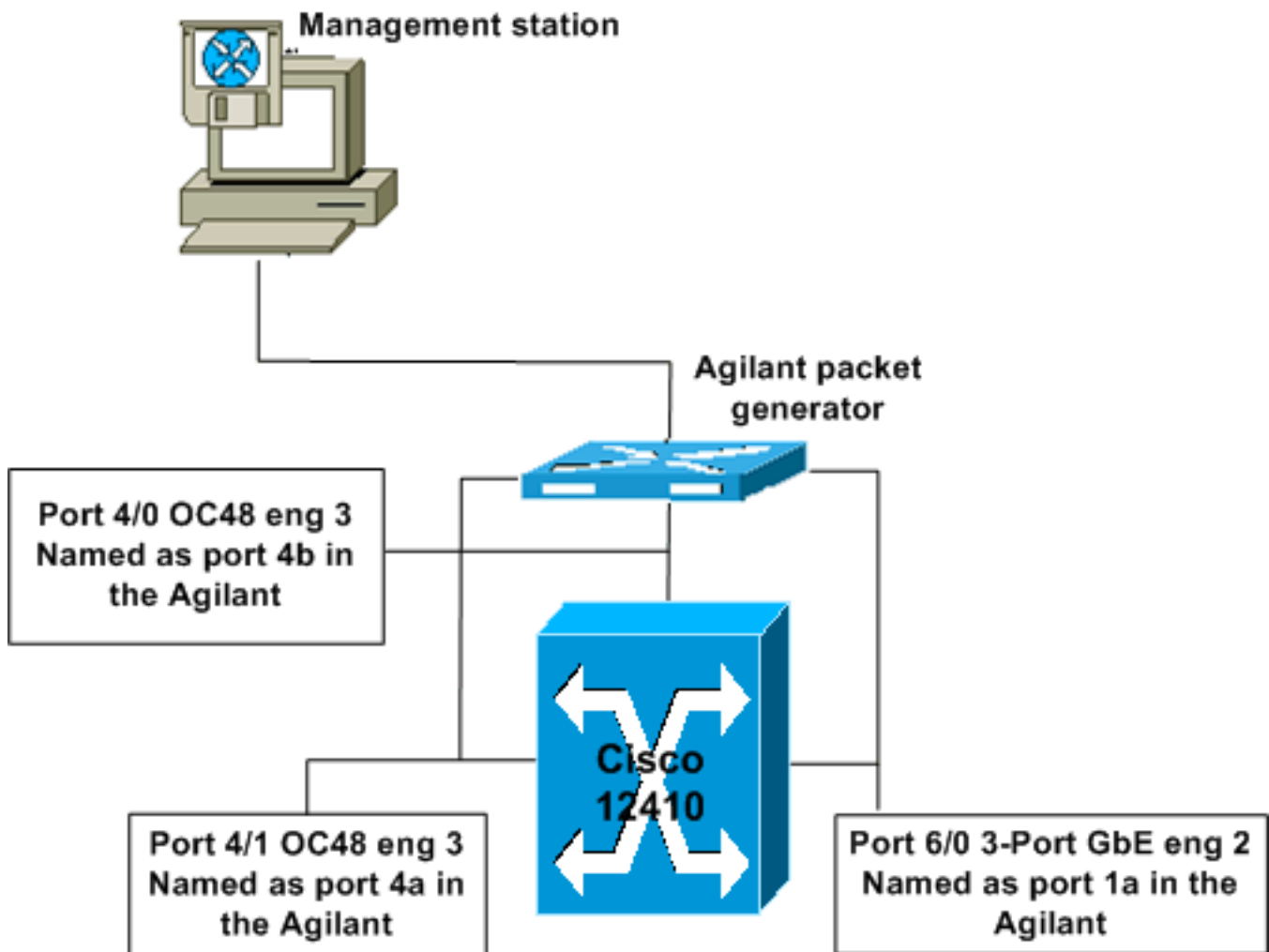
## WRED behaviour



Vous devriez n'avoir la plus basse latence possible aux paquets vocaux, et aucune baisse du tout pour la Voix et le trafic de multidiffusion.

## Configuration du réseau

Pour tester et évaluer la configuration, nous avons utilisé Cisco 12410 ainsi qu'un générateur de paquet d'Agilant. Le routeur de Cisco 12000 exécute une version d'ingénierie basée sur le Logiciel Cisco IOS version 12.0(21)S1.



## [implémentation de Mise en file d'attente d'Engine 2 du GbE 3-Port](#)

Les cartes d'Engine 2 ont normalement huit files d'attente de fromfab et une file d'attente à faible latence, et huit tofab queue par emplacement de destination. Il y a également une file d'attente distincte de Multidiffusion de tofab. Sur la carte du GbE 3-Port, il y a seulement une file d'attente de fromfab par port physique. Dans le test, la configuration qui était appliquée spécifie plus de files d'attente. Les résultats prouvent que la carte du GbE 3-Port reçoit cette configuration, et les files d'attente sont automatiquement tracées aux files d'attente qui sont disponibles.

## [Algorithme WRED d'Engine 2 en logiciel de Cisco IOS](#)

Vous devez entièrement comprendre l'algorithme utilisé pour WRED dans le linecard d'Engine 2 en configurant les valeurs de minimum et de profondeur de la file d'attente de maximum. Le code ne s'inquiète pas de la valeur minimum configurée ; au lieu de cela, il emploie sa propre formule (basée sur la valeur maximale configurée) pour placer la valeur minimum.

**Formule :**

Valeur minimum = valeur maximale - (le puissance de 2 le plus élevé qui ne génère pas un résultat négatif)

Les valeurs utilisées dans ce test ont eu comme conséquence les valeurs minimum suivantes

programmées au circuit intégré spécifique (ASIC) :

Priorité	Minimum configuré	Maximum configuré	Le puissance de 2 le plus élevé	Valeur minimum dans l'ASIC
0	50	5000	4096	5000-4096=904
1	60	6000	4096	6000-4096=1904
2	70	7000	4096	7000-4096=2904
3	80	8000	4096	8000-4096=3904

Utilisant cette formule calculer la valeur minimum signifie que vous pouvez finir par avec le comportement incorrect de traitement des paquets si vous ne prenez pas en compte ceci en configurant vos paramètres WRED. Ceci est affiché dans l'exemple suivant :

Priorité	Minimum configuré	Maximum configuré	Le puissance de 2 le plus élevé	Valeur minimum dans l'ASIC
0	50	150	128	150-128=22
1	75	225	128	225-128=97
2	100	300	256	300-256=44
3	125	375	256	375-256=119

Ceci signifie que, quoique les valeurs sont configurés pour commencer la baisse selon la règle 0 d'abord, puis 1, 2 et pour finir 3 (ci-dessus), quand les valeurs sont écrites à l'ASIC, vous commencez réellement la priorité chutante 0, puis la priorité 2, puis la priorité 1, et pour finir la priorité 3. Il n'y a aucune manière de voir quelles valeurs ont été configurées dans l'ASIC sur une carte d'Engine 2. Si vous appliquez la configuration sur une carte de l'engine 3, les valeurs apparaissant dans la configuration seront les valeurs réelles (la valeur minimum recalculée).

## Configuration QoS

Pour plus de détails au sujet de la configuration QoS, lisez s'il vous plaît [MDRR et WRED de compréhension et de configuration sur le Routeur Internet de la série Cisco 12000](#).

## Cos de Rx

```
rx-cos-slot 2 B2-Table  
rx-cos-slot 3 B2-Table  
rx-cos-slot 6 B2-Table
```

Dans la plupart des cas, vous pouvez utiliser le **rx-cos-emplacement toute la** commande. Dans notre cas de test, nous avons eu quelques cartes qui n'ont pas pris en charge la Mise en file d'attente de tofab, ainsi nous ne pourrions pas toujours utiliser le **rx-cos-emplacement toute la** commande. Au lieu de cela, nous avons assigné notre emplacement-table aux linecards étant

utilisés dans le test.

```
!  
slot-table-cos B2-Table  
destination-slot all B2  
multicast B2 !--- If you don't fulfill the steps above, you will not be able to reach the goal  
of 0 drops for Multicast traffic. With no rx-cos configured, multicast will be treated in the  
default queue, meaning it will drop as soon as there is congestion. !
```

## Tx QoS

Maintenant vous pouvez configurer votre tx-cos. Choisissez un nom pour vos qos de tx, tels que le « cos-file d'attente-groupe B2".

Chaque valeur de priorité que vous voulez configurer un comportement de baisse pour que les besoins soient tracés à une aléatoire-détecter-étiquette distincte.

```
precedence 0 random-detect-label 0  
  
precedence 1 random-detect-label 1  
  
precedence 2 random-detect-label 2  
  
precedence 3 random-detect-label 3
```

Pour le Mécanisme MDRR (Modified Deficit Round Robin), tracez chaque priorité à une file d'attente MDRR. Dans notre exemple, nous avons tracé la priorité 0-3 à la même file d'attente MDRR afin de réserver la bande passante pour le vidéo (le trafic de multidiffusion). Ce mappage fournit le comportement demandé.

```
precedence 0 queue 0  
  
precedence 1 queue 0  
  
precedence 2 queue 0  
  
precedence 3 queue 0  
  
precedence 4 queue 4
```

La Voix est identifiée par la priorité 5, qui est pourquoi la priorité 5 est tracée à la file d'attente à faible latence.

```
precedence 5 queue low-latency  
  
precedence 6 queue 6  
  
precedence 7 queue 6
```

Maintenant vous devez configurer le comportement chutant pour chacune des aléatoire-détecter-étiquettes. Pendant le test, ces nombres ont été changés jusqu'à ce qu'on ait trouvé des valeurs qui ont donné le modèle désiré de baisse. Pour des détails, voyez la section [prévue de résultats](#). La profondeur de la file d'attente est mesurée sur la file d'attente physique, et pas sur le MDRR ou la file d'attente de Rouge-étiquette.

```
random-detect-label 0 50 5000 1  
  
random-detect-label 1 60 6000 1  
  
random-detect-label 2 70 7000 1  
  
random-detect-label 3 80 8000 1
```

Sur le Cisco 12000, il est possible de créer basé sur classe, comportement de la mise en file d'attente pondérée (CBWFQ), en donnant à la file d'attente différente MDRR par poids. Le poids par défaut est 10 par file d'attente. Le nombre d'octets a transmis chaque cycle MDRR est une fonction de la valeur de poids. Une valeur de 1 signifie 1500 octets chaque cycle. Une valeur de 10 signifie  $1500 + (9 * 512)$  des octets par cycle. »

```
queue 0 20
```

```
queue 4 20
```

```
queue 6 20
```

```
!
```

## Mappage d'interface

Chaque interface doit être configurée pour WRED. Ceci est fait utilisant les commandes :

- configure terminal
- yole 6/0 d'interface
- tx-cos B2

## Commencer sans des baisses

Le flot généré utilise les valeurs suivantes à moins qu'autre chose soit énoncé :

```
MTU all three data streams 300byte, MTU voice 80byte, MTU MC 1500byte
```

```
126Mb MC, 114Mb voip
```

Ceci a comme conséquence un flot de fond de 240Mb (VoIP et Multidiffusion).

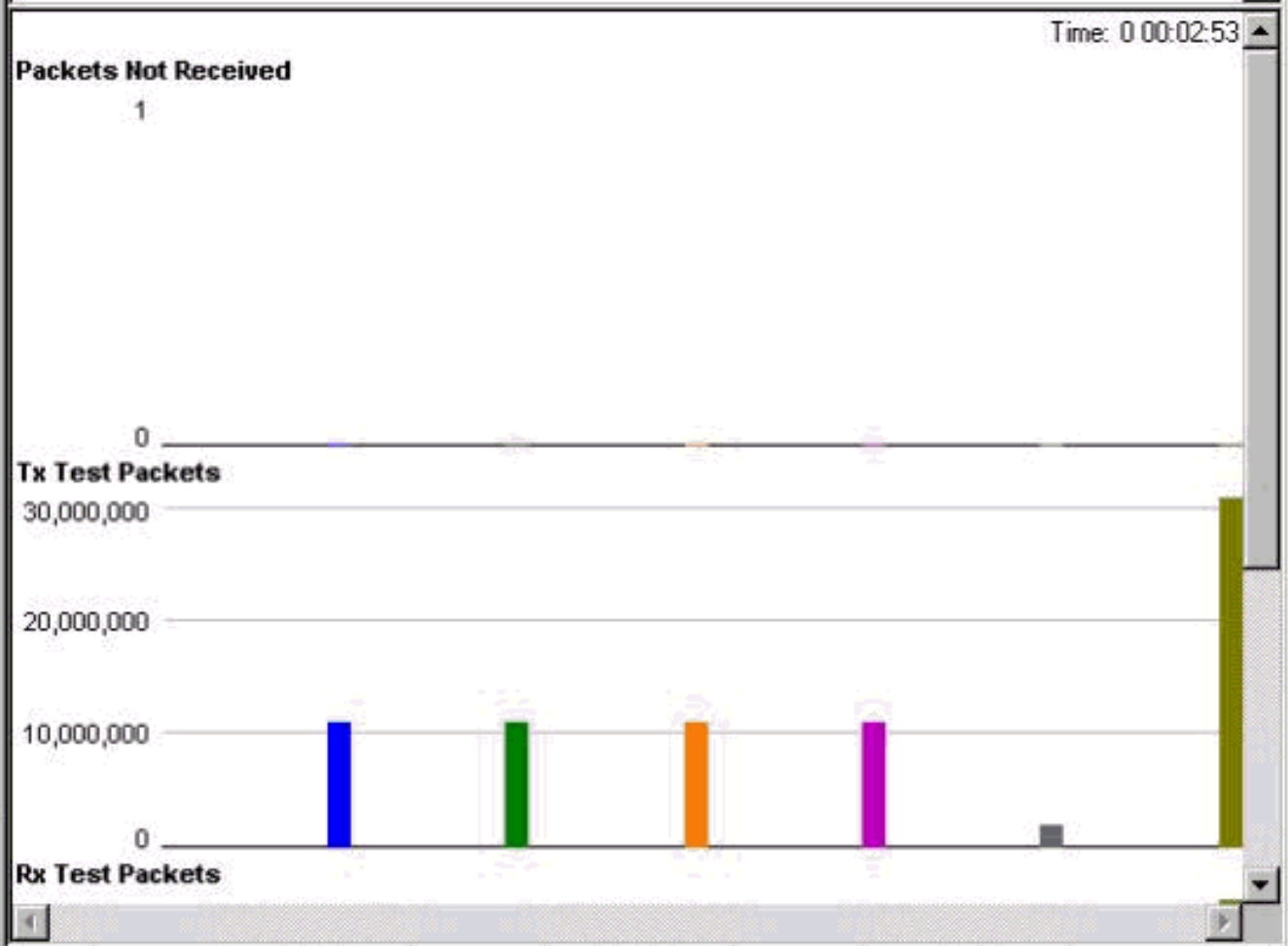
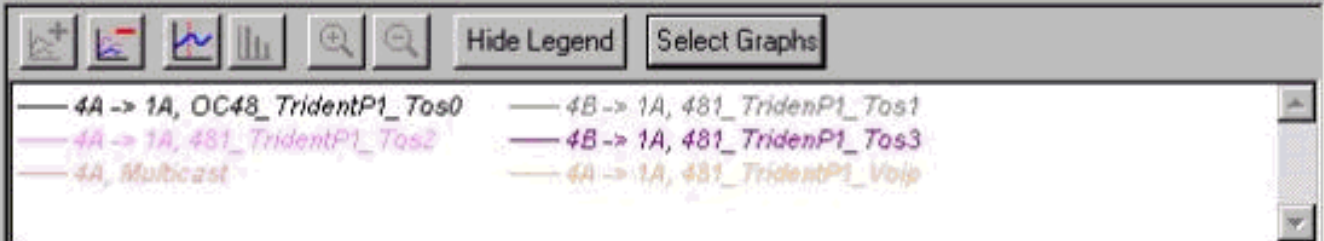
Nous ajoutons alors quatre flux de données de la même taille, mais avec la priorité 0-3 (une valeur de priorité par flot).

## Quatre flux de données de 151Mb chacun

Cette configuration donne une bande passante totale de 844Mb. Le graphique ci-dessous prouve qu'il y a les pertes de paquets 0, et une latence très basse (environ 50 nous - des microsecondes - pour chaque flot, y compris la Voix).



Ports/Streams	Seq Errors	Packets Not R...	Avera...	Max Latency...	T: ▲
- 4B -> 1A, 481_TridenP1_Tos1	0		30.39	44.36	
- 4B -> 1A, 481_TridenP1_Tos3	0		35.29	48.94	
- 4A -> 1A, 481_TridenP1_Tos2	0		33.73	48.98	
- 4A -> 1A, 0C48_TridenP1_Tos0	0		28.90	44.50	
- 4A -> 1A, 481_TridenP1_Voip	0		32.99	52.84	
+ Port 1B			51.04	56.58	
+ Port 1C			45.12	49.80	
Port 1D		0			



### [Quatre flux de données de 160Mb chacun](#)

Cette configuration donne une bande passante totale de 880Mb. Le graphique ci-dessous prouve que le début de paquets pour relâcher de la priorité 0 classes, et la latence pour la Voix a grimpé jusqu'à 6 ms - des millisecondes.

Ports/Streams	Seq Errors	Packets Not R...	Avera...	Max Latency...	T: ▲
- 4B -> 1A, 481_TridentP1_Tos1	0		1566.55	13117.44	
- 4B -> 1A, 481_TridentP1_Tos3	0		1571.17	13118.92	
- 4A -> 1A, 481_TridentP1_Tos2	0		1570.53	13124.36	
- 4A -> 1A, DC48_TridentP1_Tos0	114207	162429	1408.37	13117.96	
- 4A -> 1A, 481_TridentP1_Voip	0		815.62	6337.22	
+ Port 1B			847.10	6474.38	
+ Port 1C			834.87	6395.62	
+ Port 1D		0			

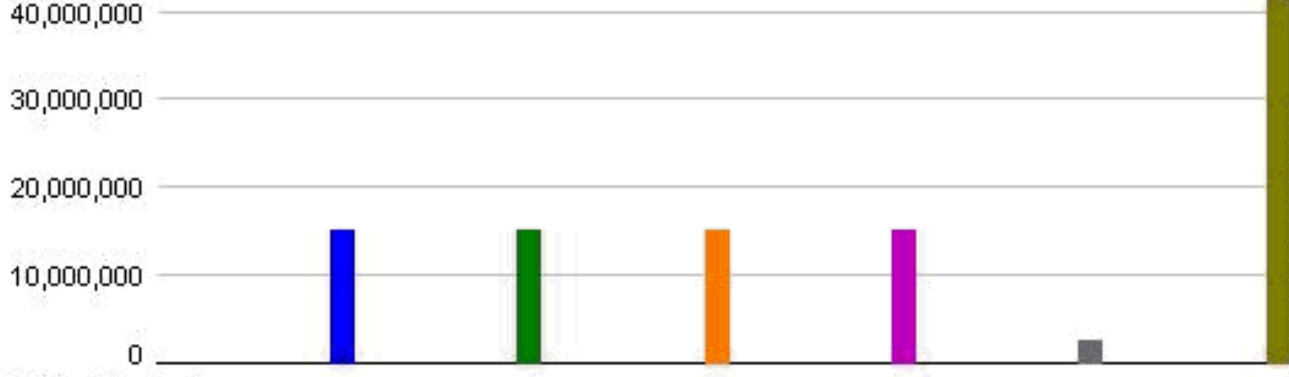
— 4A -> 1A, OC48\_TridentP1\_Tos0      — 4B -> 1A, 481\_TridentP1\_Tos1  
 — 4A -> 1A, 481\_TridentP1\_Tos2      — 4B -> 1A, 481\_TridentP1\_Tos3  
 — 4A, Multicast                              — 4A -> 1A, 481\_TridentP1\_Voip

Time: 0 00:03:54 ▲

**Packets Not Received**



**Tx Test Packets**

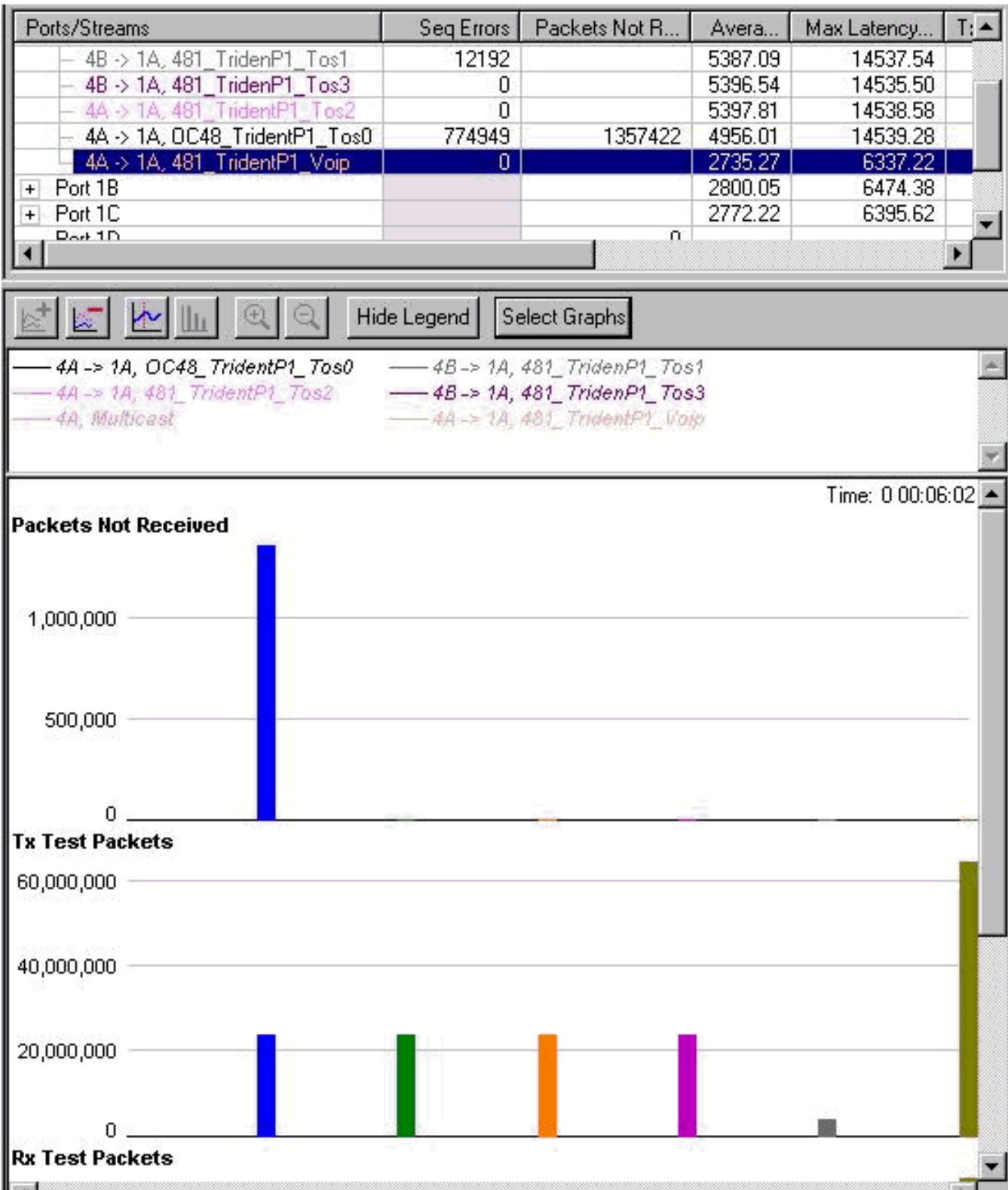


**Rx Test Packets**

[Quatre flux de données de 167Mb chacun](#)

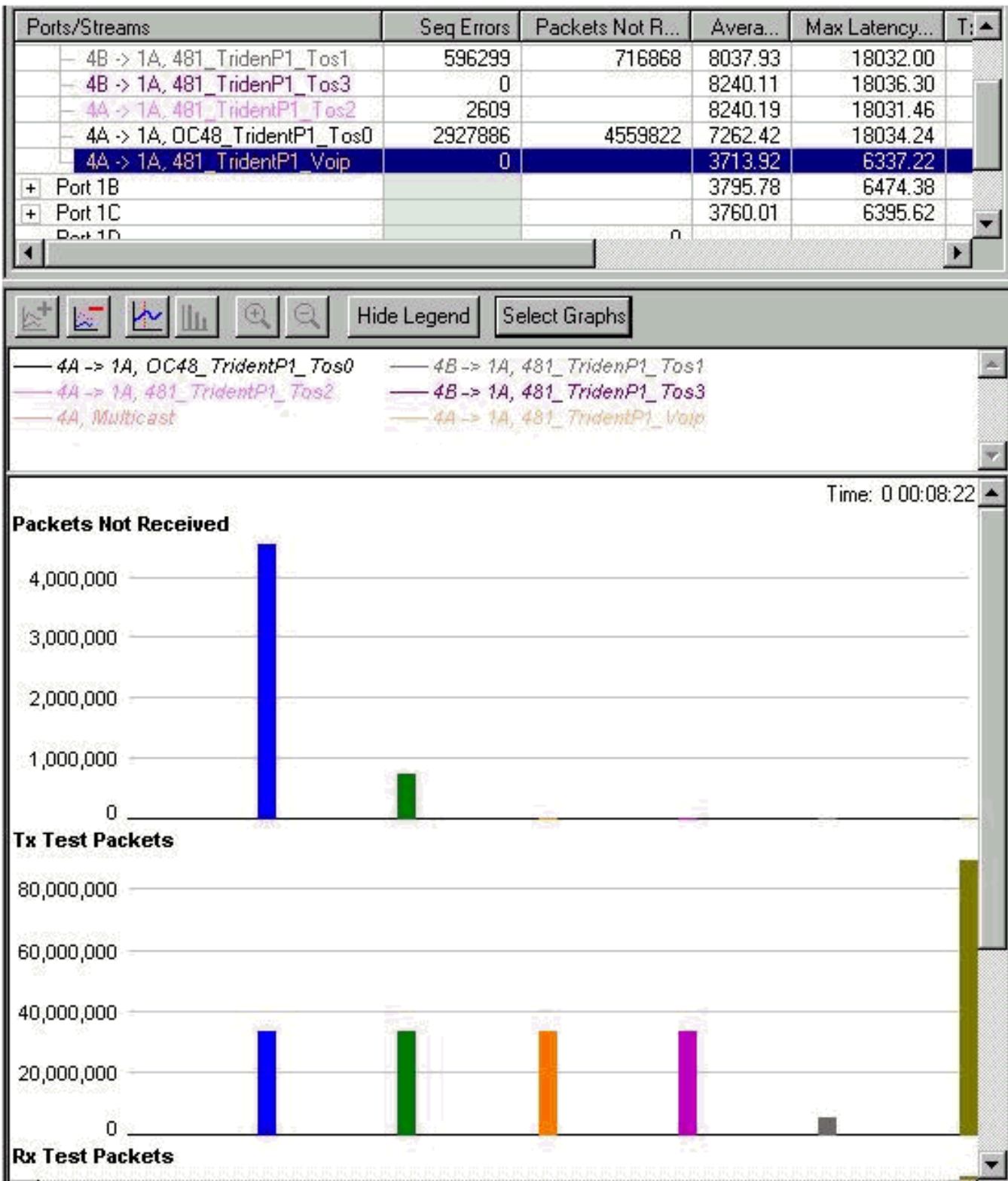
Cette configuration donne une bande passante totale de 908Mb. Les baisses commencent maintenant pour la priorité 1 classe aussi bien. La latence du trafic vocal est toujours identique.

**Remarque:** Le flot n'a pas été arrêté avant d'être augmentée, ainsi la différence entre le nombre de baisses dans le flot 0 et 1 est cumulative.



### [Quatre flux de données de 191Mb chacun](#)

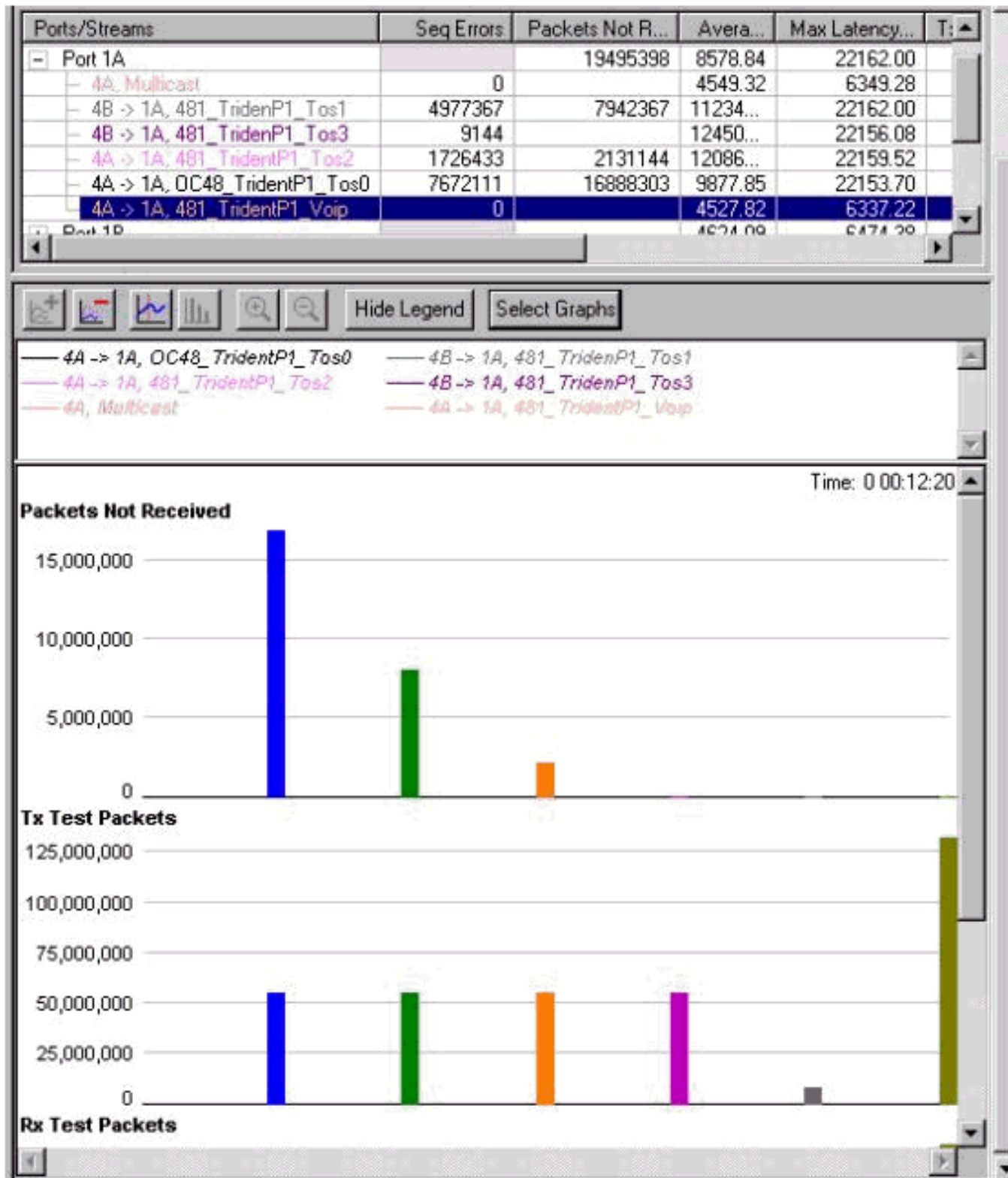
Quand la bande passante totale augmente, les paquets commencent à relâcher de la file d'attente de la priorité 2 aussi bien. La bande passante totale que nous essayons d'atteindre pour l'interface de Gigabit Ethernet est maintenant 1004Mb. Ceci est illustré dans les compteurs d'erreurs d'ordre dans le graphique ci-dessous.



### Quatre flux de données de 244Mb chacun

Les erreurs d'ordre pour la priorité 3 commencent à augmenter aussi bien. C'est le premier signe que les baisses commenceront à partir de cette file d'attente. La bande passante totale que nous essayons d'envoyer l'interface de GbE est maintenant la mi-bande 1216. Notez que les baisses sur la Multidiffusion (MC) et la file d'attente de Voix sont encore zéro, et la latence de la file d'attente de Voix n'a pas augmenté.

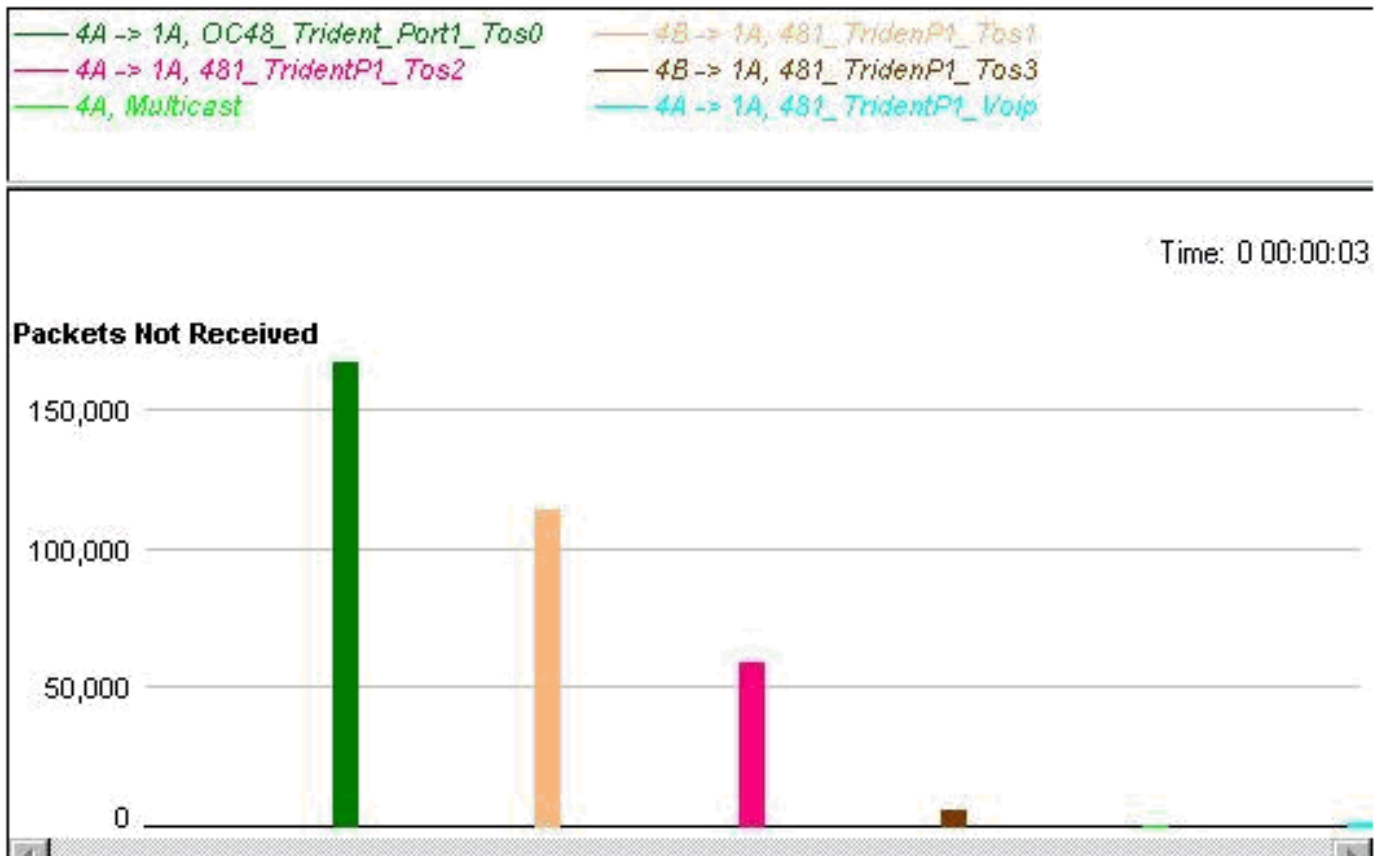




### Arrêter et commencer

Tous les flots ont été arrêtés et ont commencé à générer un graphique qui a effacé des compteurs. Ceci affiche comment il regardera pendant l'encombrement lourd. Comme vous pouvez voir ci-dessous, le comportement est désiré.

Ports/Streams	Packets Not Received
All Ports	244885
[-] Port 1A	259941
[-] 4A -> 1A, OC48_Trident_Port1_Tos0	133621
[-] 4A -> 1A, 481_TridentP1_Tos2	45024
[-] 4A -> 1A, 481_TridentP1_Voip	
[-] 4A, Multicast	
[-] 4B -> 1A, 481_TridentP1_Tos1	89282
[-] 4B -> 1A, 481_TridentP1_Tos3	624



## Tout le QoS a retiré

Pour montrer que QoS améliore vraiment la représentation pendant l'encombrement, QoS a été maintenant retiré et l'interface a été congestionnée. Les résultats sont ci-dessous (le flot généré utilise les valeurs suivantes à moins qu'autre chose soit énoncé).

MTU all three data streams 300byte, MTU voice 80byte, MTU MC 1500byte

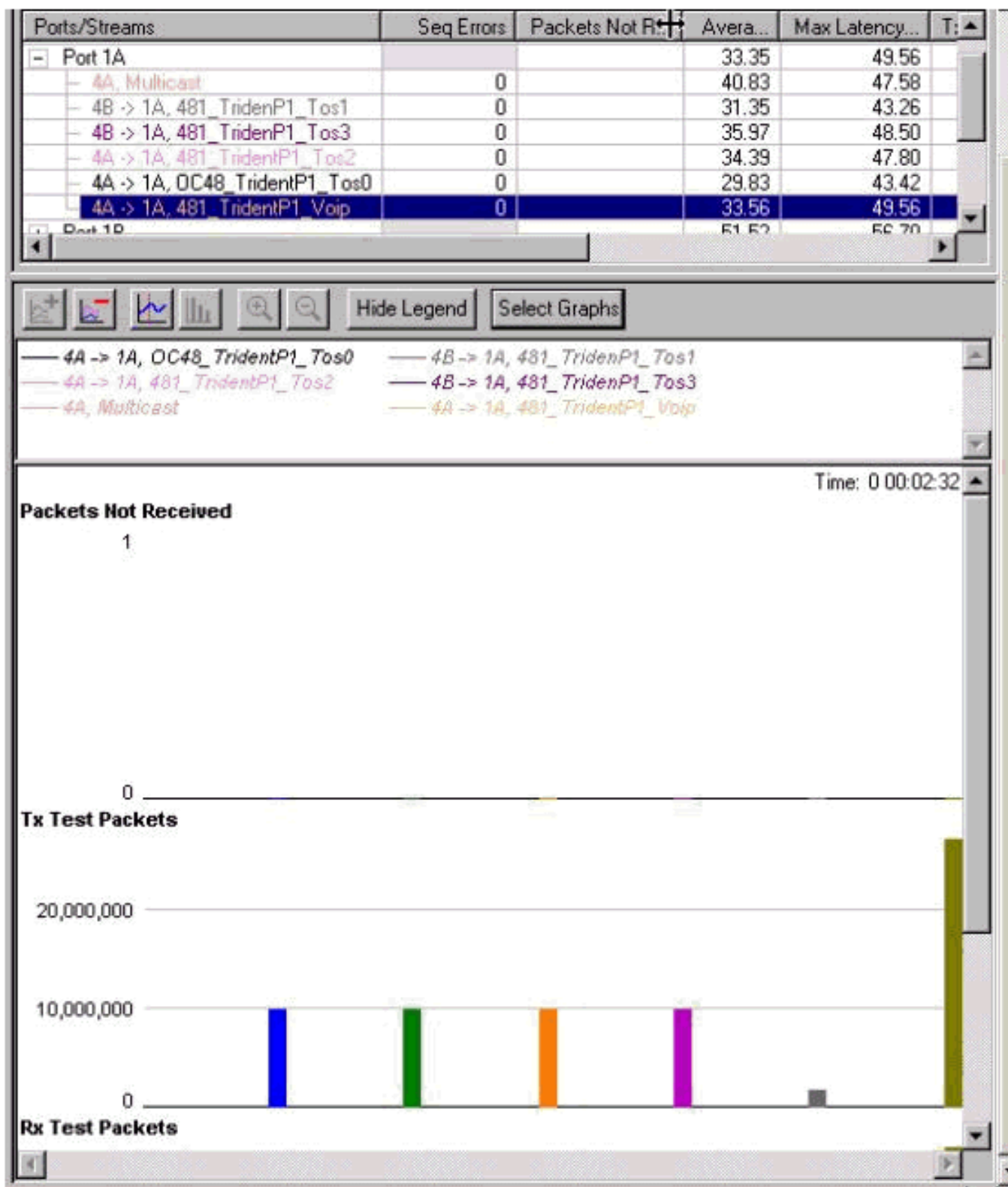
126Mb MC, 114Mb VoIP

Ceci a comme conséquence un flot de fond de 240Mb (VoIP et Multidiffusion).

Nous ajoutons alors quatre flux de données de la même taille, mais avec la priorité 0-3 (une valeur de priorité par flot).

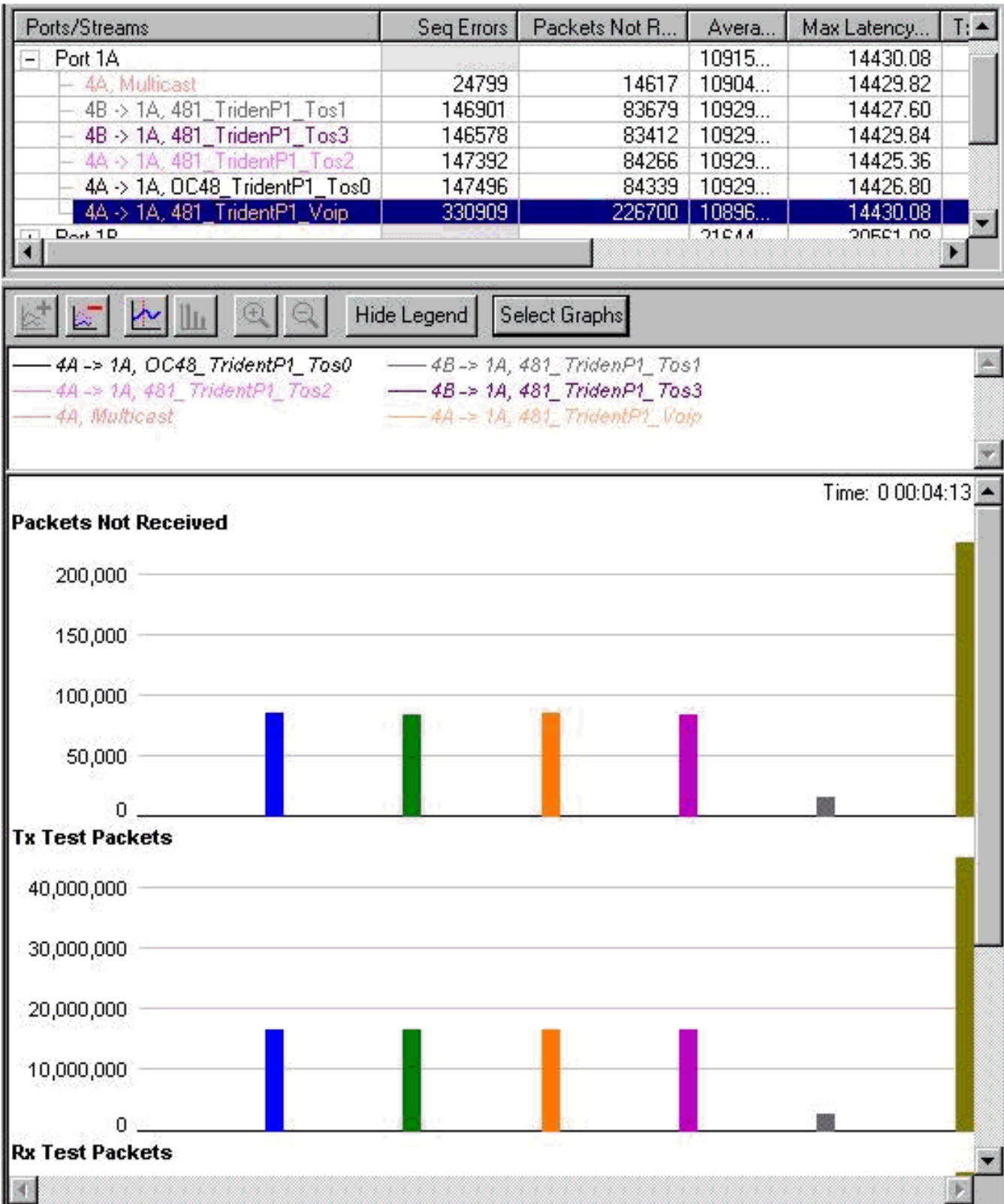
## Quatre flux de données de 153Mb chacun

Ceci donne un total de 852Mb. Il y a les gouttes 0, et une latence de moins puis 50 nous.



### [Quatre flux de données de 158Mb chacun](#)

Nous commençons à relâcher à l'utilisation à peu près identique comme quand WRED est appliqué (872Mb). La différence est maintenant qu'il y a une latence des paquets vocaux de 14 ms (plus de deux fois plus que dans le test WRED), et que les baisses se produisent également de toutes les classes, y compris le VoIP et la Multidiffusion.



## Ajouter le chargement d'entrée

Jusqu'ici, tous les tests ont seulement inclus la transmission par les interfaces de Gigabit Ethernet. Pour vérifier comment l'interface réagit dans une situation où nous congestionnons également l'interface dans l'autre direction, les tests suivants ont été faits.

Pour ce test, nous avons chargé l'interface de Gigabit Ethernet avec une quantité totale de bande 1056. Ceci a eu comme conséquence les baisses sur la priorité 0-2, aucune baisses sur le trafic de la priorité 3. (MC et VOIP sont restés le même, c.-à-d., aucune baisses). Nous avons



alors ajouté le trafic dans l'autre direction, autant le trafic que le générateur de paquet pouvait envoyer par l'interface de Gigabit Ethernet. Le résultat est assez impressionnant : l'encombrement de réception ne gêne pas du tout la file d'attente de transmission, et la latence pour le trafic de réception est extrêmement - bas, moins de 13 nous pour la Voix.

- Port 1A	21976403	12050.70	20237.92
- 4A, Multicast		6277.06	6402.88
- 4B -> 1A, 481_TridenP1_Tos1	7729260	16763.90	20232.04
- 4B -> 1A, 481_TridenP1_Tos3		17287.73	20237.92
- 4A -> 1A, 481_TridenP1_Tos2	2090730	17139.98	20233.44
- 4A -> 1A, OC48_TridenP1_Tos0	17210053	16519.08	20236.76
- 4A -> 1A, 481_TridenP1_Voip		6245.21	6376.02
+ Port 1B		6379.18	6512.30
+ Port 1C		6323.52	6432.94
Port 1D	0		
- Port 4A	7	13.09	15.02
- 1A -> 4A, 481_Triden_P3_tos3		13.37	15.02
- 1A -> 4A, 481_TridenP3_Voip		12.67	14.30
- 1A -> 4A, 48_TridenP3_tos2		13.23	14.94
- Port 4B	6	13.11	14.64
- 1A -> 4B, 48_Triden_P3_Tos0		13.11	14.62
- 1A -> 4B, 481_TridenP3_Tos1		13.11	14.64

Ports/Streams	Tx Test Throughput (Mb/s)	Rx Test Throughput (Mb/s)
All Ports		
- Port 1A	747.47	858.75
- 4A, Multicast	126.24	126.24
- 4B -> 1A, 481_TridenP1_Tos1	189.55	142.03
- 4B -> 1A, 481_TridenP1_Tos3	189.54	189.54
- 4A -> 1A, 481_TridenP1_Tos2	189.54	175.70
- 4A -> 1A, OC48_TridenP1_Tos0	189.54	95.24
- 4A -> 1A, 481_TridenP1_Voip	130.00	130.00
+ Port 1B	0.00	126.24
+ Port 1C	0.00	126.24
Port 1D	0.00	0.00
- Port 4A	635.32	424.24
- 1A -> 4A, 481_Triden_P3_tos3	161.62	161.62
- 1A -> 4A, 481_TridenP3_Voip	101.01	101.01
- 1A -> 4A, 48_TridenP3_tos2	161.62	161.62
- Port 4B	379.09	323.23
- 1A -> 4B, 48_Triden_P3_Tos0	161.62	161.62
- 1A -> 4B, 481_TridenP3_Tos1	161.62	161.62

## [12-RP#show relie g 6/0](#)

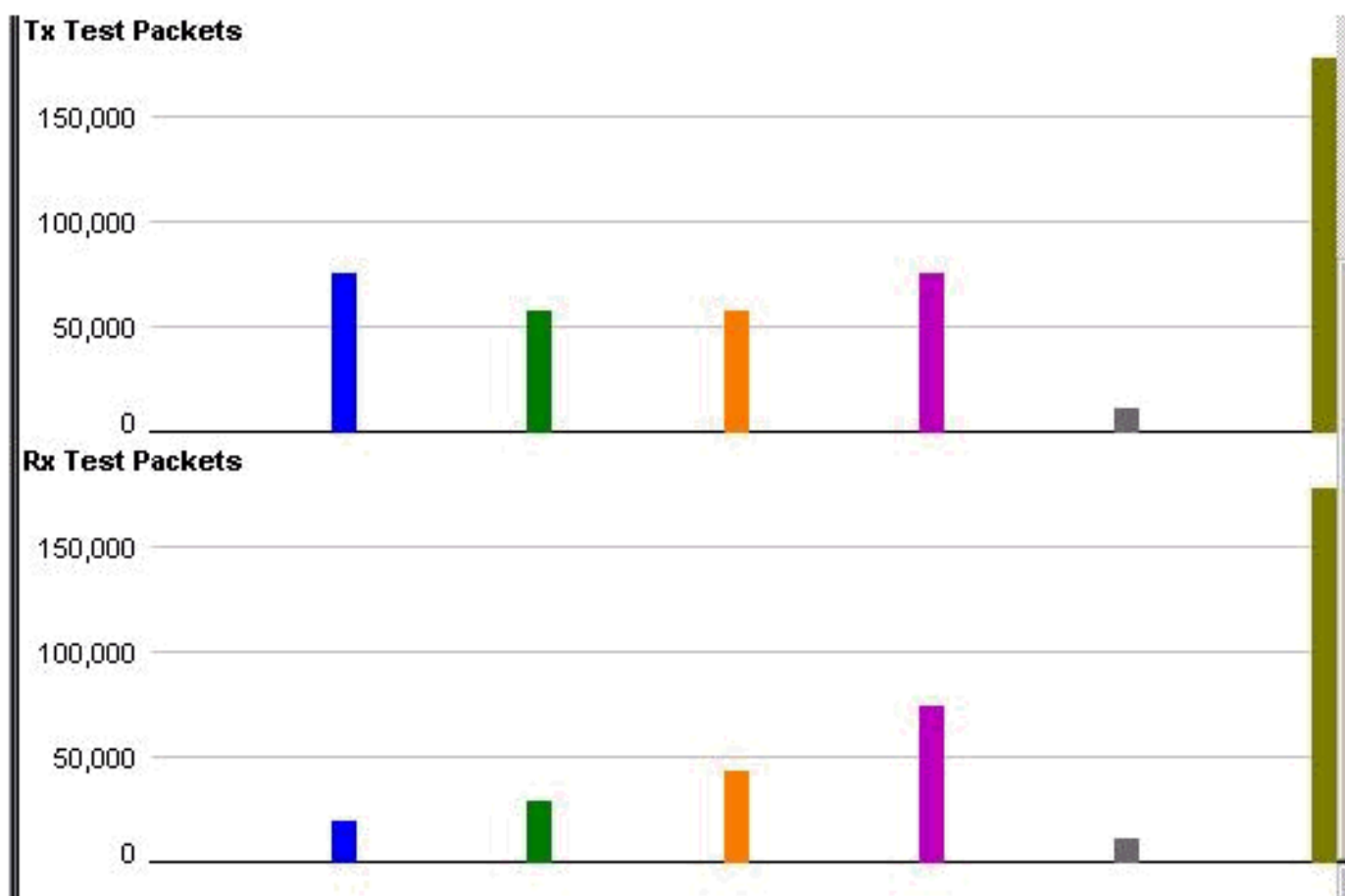
Vous pouvez surveiller le chargement sur la liaison Gigabit utilisant la commande d'interfaces d'exposition :

```
Router#show interfaces gig 6/0 GigabitEthernet6/0 is up, line protocol is up Hardware is GigMac
3 Port GigabitEthernet, address is 0004.de56.c264 (bia 0004.de56.c264) Internet address is
178.6.0.1/24 MTU 1500 bytes, BW 1000000 Kbit, DLY 10 usec, rely 255/255, load 232/255
Encapsulation ARPA, loopback not set Keepalive set (10 sec) Full-duplex mode, link type is
force-up, media type is SX output flow-control is unsupported, input flow-control is off ARP
```

```
type: ARPA, ARP Timeout 04:00:00 Last input 00:00:05, output 00:00:05, output hang never Last clearing of "show interface" counters 08:52:40 Queueing strategy: random early detection (WRED) Output queue 0/40, 2174119522 drops; input queue 0/75, 0 drops 30 second input rate 838969000 bits/sec, 792079 packets/sec 30 second output rate 910819000 bits/sec, 464704 packets/sec 7584351146 packets input, 1003461987270 bytes, 0 no buffer Received 0 broadcasts, 0 runts, 0 giants, 0 throttles 0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored 0 watchdog, 514 multicast, 0 pause input 11167110605 packets output, 2241229569668 bytes, 0 underruns 0 output errors, 0 collisions, 0 interface resets 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
```

## Changer la taille des flots

Pour vérifier que les résultats de test ne sont pas dus à la bande passante étant identique pour tous les flots, nous avons changé les flots de sorte qu'ils aient transmis différentes quantités de données. Nous avons également essayé de changer le Maximum Transmission Unit (MTU) ainsi il était différent pour chaque flot. Avec les valeurs configurées de file d'attente, le résultat était toujours identique, d'abord, puis relâchant la priorité 0 1, puis 2, et pour finir priorité 3.



## Vérifier avec un linecard de l'engine 4 des Gigabit Ethernet 10-Port

Puisque la latence de la file d'attente VoIP (la file d'attente à faible latence) dans le test était assez élevée, nous avons réalisé le même essai avec le linecard de l'engine 4 des Gigabit Ethernet 10-Port. Comme prévu, le résultat avec ce linecard était bien mieux concernant la latence dans la file d'attente à faible latence (LLQ). Les résultats étaient identiques quant à la façon dont la baisse s'est produite. La latence pour le LLQ était autour de 10us, qui est 1:1000 de la latence dans le linecard d'Engine 2 des Gigabit Ethernet 3-Port.

## Informations connexes

- [Compréhension et configuration de MDRR et de WRED sur le Routeur Internet de la série Cisco 12000](#)
- [Comment lire la sortie de la commande show controller frfab | Commandes tofab queue sur un routeur Internet de la gamme Cisco 12000](#)
- [Type de service dans l'ensemble des protocoles IP \(Internet Protocol\), priorité \(RFC-1349\)](#)
- [Dépistage précoce aléatoire pesé](#)
- [Support et documentation techniques - Cisco Systems](#)