

Paquetage QOS - Qualité de service

Version 3.10.5

Frank Meyer
courriel: frank@fli4l.de

L'équipe fli4l
courriel: team@fli4l.de

16 février 2016

Table des matières

1. Documentation du paquetage QOS	3
1.1. QoS - Qualité de service	3
1.1.1. Configuration	3
1.1.2. Applications et exemples	10
A. Annexe du paquetage QOS	18
Table des figures	19
Liste des tableaux	20
Index	21

1. Documentation du paquetage QOS

1.1. QoS - Qualité de service

Grâce à Qos, vous pouvez régler et partager la bande passante disponible, sur les différents ports, adresses IP et d'autres support.

Le modem gère le Packet-Queue (ou la file d'attente) (Queue = file, en serie (être dans la file)) pour le stockage des paquets, quand la bande passante du modem est inférieur à la quantité de paquet disponible. Pour les modems DSL, la bande passante est plus grande. Ils ont l'avantage d'être relativement homogène et exploite au maximum de bande passante. Le routeur envoie au modem les paquets dans un court laps de temps (très court), le modem doit stockés ces paquets dans une file d'attente pour les traités. Ainsi, cette file d'attente a une organisation très simple, tous les paquets sont rangés et envoyés dans l'ordre d'arrivés, c'est juste un modem :-D

C'est ici que QoS entre en jeux. QoS gère également une file d'attente pour les paquets, dans le routeur lui-même, et là, on a la possibilité d'être le roi, on décide si les paquets doivent être envoyés en premier ou en second. Si QoS est configuré correctement, les paquets seront envoyés judicieusement et rapidement au modem sans qu'il n'attérissent dans la file d'attente du modem, s'est comme si on avait déplacé la file d'attente du modem sur le routeur.

Encore une chose au sujet des unités de vitesse commune : QoS prend en charge les Mibit/s (megabit/s) et les Kibit/s (kilobit/s) et surtout 1Mibit = 1024Kibit.

1.1.1. Configuration

OPT_QOS Vous pouvez placer 'yes' dans la variable OPT_QOS pour activer QoS ou 'no' pour ne pas l'utiliser

QOS_INTERNET_DEV_N Vous indiquez ici Le nombre d'interface du routeur, qui envoie les données sur Internet.

QOS_INTERNET_DEV_x Vous indiquez ici, la liste des interfaces sur lesquels les données seront transmis vers Internet. Exemples :

QOS_INTERNET_DEV_N='3'	Nombre de périphérique
QOS_INTERNET_DEV_1='ethX'	Pour cable et autres liaison Ethernet
QOS_INTERNET_DEV_2='ppp0'	Pour DSL sur protocole PPPoE
QOS_INTERNET_DEV_3='ipppX'	Pour ISDN

Le premier circuit du périphérique ISDN devrait s'appeler ippp0, et le deuxième circuit ippp1 etc ... Mais, si pour le premier circuit l'agrégation des canaux a été activée (2 canaux par circuit), alors du premier circuit avec deux canaux s'appellera ippp1 et le deuxième Circuit s'appellera ippp2. Il convient pour bénéficier du QoS avec ISDN, de désactiver l'agrégation des canaux des circuits.

QOS_INTERNET_BAND_DOWN On indique ici la bande passante maximum montante pour un accès Internet. Voir ci-dessus pour [les unités de vitesse](#) (Page 3).

Remarque : pour l'actualisation des tâches, comme pour l'affectation des paquets-ACK, il ne faut pas indiquer une quantité de bande passante (ou B-P) supérieure à la B-P vraiment disponible, les paquets de la file d'attente du routeur seront triés comme il se doit, mais cela ne sera pas tout à fait exacte, et finalement les paquets seront stockés dans la file d'attente du modem et sera ralenti. Il est possible que le fournisseur d'accès n'indique pas réellement la bande passante correspondante, cela peut être un peu plus ou un peu moins. Il faut donc faire des tests. Pour trouver la bonne quantité.

QOS_INTERNET_BAND_UP On indique ici la bande passante maximum descendante pour l'accès Internet. Voir OPT_QOS, pour les unités de vitesse.

Remarque : voir les informations ci-dessus QOS_INTERNET_BAND_DOWN.

QOS_INTERNET_DEFAULT_DOWN On indique ici les classes par défaut des paquets qui proviennent d'Internet. Tous les paquets qui n'ont pas été paramétrés dans les classes filtrées, atterrissent ensuite dans la classe supplémentaire indiqué ici.

Si aucune classe n'est paramètre et si la variable est sur

```
QOS_CLASS_x_DIRECTION='down'
```

alors, on indiquera :

```
QOS_INTERNET_DEFAULT_DOWN='0'
```

Exemple :

On met en place 2 classes et un seul filtre pour les paquets. On paramètre par ex. une adresse-IP spécifique à la 1ère des deux classes. Tous les autres paquets passeront par la 2ème classe, c'est une classe par défaut qui ne sera pas filtrée. On indique par conséquent dans la variable :

```
QOS_INTERNET_DEFAULT_DOWN='2'
```

Il faut faire attention à se que la variable QOS_INTERNET_DEFAULT_DOWN a une classe de paramétrée et que dans la variable QOS_CLASS_x_DIRECTION le paramètre 'down' soit indiqué.

QOS_INTERNET_DEFAULT_UP On indique ici les classes par défaut, pour des paquets qui vont sur Internet. Tous les paquets qui n'ont pas été paramétrés dans les classes filtrées, atterrissent ensuite dans la classe supplémentaire indiqué ici.

Si aucune classe n'est paramètre et si la variable est sur

```
QOS_CLASS_x_DIRECTION='up'
```

alors, on indiquera :

```
QOS_INTERNET_DEFAULT_UP='0'
```

L'ensemble des fonctions sont similaire à la variable QOS_INTERNET_DEFAULT_DOWN.

Il faut faire attention à se que la variable QOS_INTERNET_DEFAULT_UP a une classe de paramétrée et que dans la variable QOS_CLASS_x_DIRECTION le paramètre 'up' soit indiqué.

QOS_CLASS_N On indique ici le nombre souhaité de classe.

QOS_CLASS_x_PARENT Dans cette variable, on peut superposer les classes. On indique toujours un numéro pour la classe parent et on attribue la bande passante sur la classe parent, elle peut aussi être répartie en sous-classe. Le niveau maximal de d'imbrication s'élève à 8 niveaux, l'interface 0 représente déjà 1 niveau, il reste donc un maximum de 7 niveaux configurable

Si dans la classe il n'y a pas de sous-classe on indique :

```
QOS_CLASS_x_PARENT='0'
```

suivant la direction dont elle fait partie (avec QOS_CLASS_x_PORT) ou avec QOS_CLASS_x-PORT_TYPE et aussi la bande passante attribuée avec QOS_INTERNET_BAND_DOWN

Important : Si vous n'avez pas indiqué '0', il est important de s'assurer, que la classe se trouve bien entendu dans le niveau le plus haut (par rapport à la numérotation des niveaux).

QOS_CLASS_x_MINBANDWIDTH Vous indiquez ici bande passante minimum, que l'on veut attribuer aux classes. On peut parler aussi de rapport. Voir les indication sur les unités de vitesse dans OPT_QOS.

Exemple de classe, donc vous voulez limiter la bande passante à 128Kibit/s :

```
QOS_CLASS_1_PARENT='0'  
QOS_CLASS_1_MAXBANDWIDTH='128Kibit/s'  
QOS_CLASS_1_MINBANDWIDTH='128Kibit/s'
```

Maintenant on dispose de 3 classes supplémentaires, on paramètre les variables QOS_CLASS_x_MINBANDWIDTH et QOS_CLASS_x_MAXBANDWIDTH des sous-classes de notre première classe :

```
QOS_CLASS_2_PARENT='1'  
QOS_CLASS_2_MINBANDWIDTH='60Kibit/s'  
QOS_CLASS_2_MAXBANDWIDTH='128Kibit/s'  
  
QOS_CLASS_3_PARENT='1'  
QOS_CLASS_3_MINBANDWIDTH='40Kibit/s'  
QOS_CLASS_3_MAXBANDWIDTH='128Kibit/s'  
  
QOS_CLASS_4_PARENT='1'  
QOS_CLASS_4_MINBANDWIDTH='28Kibit/s'  
QOS_CLASS_4_MAXBANDWIDTH='128Kibit/s'
```

Toutes les sous-classes peuvent avoir la même (ou pas) priorité (voir QOS_CLASS_x_PRIO). Maintenant un débit respectif est produit dans chacune des 3 classes avec la variable QOS_CLASS_x_MINBANDWIDTH. Ainsi on attribut dans chaque classe la bande passante correspondante dans QOS_CLASS_x_MINBANDWIDTH. Exemple si vous modifier la classe 2 et que vous lui attribuez 20Kibit/s, il restera donc 40Kibit/s à attribuer aux "autres" classes. Cette excédent peut être réparti avec la relation 40/28 dans les classes 3 et 4. Avec la variable QOS_CLASS_x_MAXBANDWIDTH chaque classe sera limité à 128Kibit/s, étant donné que toutes les sous-classes sont limitées à 128Kibit/s, ils ne pourront pas dépasser le débit autorisé de 128Kibit/s.

QOS_CLASS_x_MAXBANDWIDTH Vous indiquez ici bande passante maximum, que l'on veut attribuer aux classes. Il n'y a pas sens, d'indiquer une valeur plus basse que la variable QOS_CLASS_x_MINBANDWIDTH. Si l'on indique aucune valeur, la variable prend automatiquement la valeur qui est indiqué dans QOS_CLASS_x_MINBANDWIDTH. bien entendu pour cette classe il ne faut pas indiquer un surplus de bande passante.

Voir OPT_QOS pour les unités de vitesse.

QOS_CLASS_x_DIRECTION On indique dans cette variable la direction de la bande passante, donc la classe fait partie. Si elle fait parti du débit montant, on indique :

```
QOS_CLASS_x_DIRECTION='up'
```

Si elle fait parti du débit descendant, on indique :

```
QOS_CLASS_x_DIRECTION='down'
```

Attention : Up-stream (débit montant) c'est la transmission de votre ordinateur au serveur-Internet et Down-stream (débit descendant) c'est la transmission du serveur-Internet à votre ordinateur.

QOS_CLASS_x_PRIO Dans cette variable on règle le niveau de priorité de la classe. Plus le chiffre est bas plus le niveau de priorité est élevé. Les chiffres autorisés sont de 0 à 7. Si la variable est laissée vide le niveau de priorité sera à 0 donc le plus haut.

Lorsqu'il y a un excédent de bande passante, le système détermine la classe et ça priorité pour augmenter le débit. Pour expliquer cela, nous allons modifier légèrement l'exemple QOS_CLASS_x_MINIMUMBANDWIDTH, la première classe rien n'a été changée. On attribue à la classes 4 une priorité 2 :

```
QOS_CLASS_2_PARENT='1'
QOS_CLASS_2_MINBANDWIDTH='60Kibit/s'
QOS_CLASS_2_MAXBANDWIDTH='128Kibit/s'
QOS_CLASS_2_PRIO='1'

QOS_CLASS_3_PARENT='1'
QOS_CLASS_3_MINBANDWIDTH='40Kibit/s'
QOS_CLASS_3_MAXBANDWIDTH='128Kibit/s'
QOS_CLASS_3_PRIO='1'

QOS_CLASS_4_PARENT='1'
QOS_CLASS_4_MINBANDWIDTH='28Kibit/s'
QOS_CLASS_4_MAXBANDWIDTH='128Kibit/s'
QOS_CLASS_4_PRIO='2'
```

Dans cette exemple si la classe 2 ne consomme que 20Kibit/s, il y a donc un excédent de 40Kibit/s. les Classes 3 et 4 peuvent recevoir encore plus de bande passante. Cependant, la classe 3 a une priorité plus élevée que la classe 4, donc la classe 3 peut récupérer l'excédent de la bande passante les 40Kibit/s.

Cependant si la classe 3 a besoin seulement de 20Kibit/s de l'excédent des 40Kibit/s, alors la classe 4 recevra les 20Kibit/s restant.

Prenons un autre exemple, si la classe 4 ne consomme pas la bande passante et si les classes 2 et 3 on besoin de plus de bande passante. Alors, chaque classes utilise la bande passante spécifiée dans la variable QOS_CLASS_x_MINBANDWIDTH le reste sera divisé entre les deux avec le rapport 60/40, puisque les deux classes ont la même priorité.

La variable QOS_CLASS_x_PRIO influe seulement sur l'excédent de bande passante, pour éventuellement là répartir.

QOS_CLASS_x_LABEL Avec cette variable optionnelle, vous pouvez placer un intitulé pour une classe. Cet intitulé sera affiché pour le graphique de QOS dans OPT_RRDTOOL s'il est activé.

QOS_FILTER_N Vous indiquez ici le nombre de filtre souhaité.

Au sujet du filtrage, en général on peut dire ceci : les paramètres des différentes variables sont reliées et/ou les différentes paramètres d'une même variable sont aussi reliées. on

peut dire par exemple : que dans un même filtre nous pouvons filtrer une adresse IP et un port, ainsi seul des paquets filtrés seront envoyés vers la cible de la (n) classe, et seront appliqués sur l'un et l'autre paramètre simultanément.

Un autre exemple : Dans le même filtre il y a deux ports (21 et 80) et une adresse IP. Bien sûr, un paquet de données ne peut pas être envoyé sur les deux ports en même temps. Le filtre se comporte alors comme ceci : les paquets filtrés utilisent le port 21 et en même temps l'adresse IP, ou le port 80 et en même temps l'adresse IP.

Important : Cela dépend de la séquence de filtrage !

Un exemple : on veut transporter les paquets, par l'intermédiaire du port 456 qui est ouvert, pour **tous** les ordinateurs de la classe A. De plus on voudrait faire passer tous les paquets de l'ordinateur de l'adresse IP 192.168.6.5 - par le port 456 ouvert - dans la classe B. Je règle maintenant le filtre de la première adresse IP, alors tous les paquets arrivent - sur le port 456 ouvert - dans la classe B, je règle un autre filtre pour le port 456 qui n'est pas altéré. Le filtre pour le port 456 doit donc être avant le filtre de l'adresse IP 192.168.6.5

QOS_FILTER_x_CLASS On indique dans cette variable, les classes qui doivent s'appliquer aux paquets filtrés. Par exemple, on veut filtrer les paquets de la variable spécifié ici QOS_CLASS_25_MINBANDWIDTH avec le numéro de classe, vous devez indiquer :

```
QOS_FILTER_x_CLASS='25'
```

Avec la variable QOS_CLASS_x_DIRECTION on peut indiquer la classe appartenant maintenant au débit montant ou au débit descendant. Si maintenant on met en place un filtre pour filtrer les paquets par exemple dans la classe du débit montant, seul des paquets du débit montant seront ainsi filtrés par ce filtre et seront installés pour la classe donnée. Avec cette variable QOS_CLASS_x_DIRECTION on détermine la "direction" du débit à filtrer. Depuis la version 2.1, il est désormais possible de l'indiquer plus d'une classe. Par exemple si vous souhaitez, envoyer le trafic sur le port 456 à la fois dans les classes des débit montant et descendant, vous pouvez indiquer :

```
QOS_FILTER_x_CLASS='4 25'
```

Avec les numéros de classe 4 pour le débit montant et de classe 25 pour le débit descendant. Il n'y a aucun sens à indiquer ici un débit montant et descendant, cela est purement objectif à ne pas recopier.

QOS_FILTER_x_IP_INTERN On indique dans cette variable les adresses-IP et les adresses de domaine du réseau interne, qui doivent être filtrés. Elles sont séparées par un espace et peuvent être combinées librement.

Cela pourrait être par exemple :

```
QOS_FILTER_x_IP_INTERN='192.168.6.0/24 192.168.5.7 192.168.5.12'
```

Ici, toutes les adresses sous la forme 192.168.6.X sont filtrées en plus des adresses IP 192.168.5.7 et 192.168.5.12.

Cette variable peut aussi rester vide.

Si vous utilisez cette variable QOS_FILTER_x_IP_EXTERN avec une adresse-IP ou une plage d'adresses-IP, il y aura pas de trafic filtré entre QOS_FILTER_x_IP_INTERN et QOS_FILTER_x_IP_EXTERN.

Important: Si vous ajoutez la variable `QOS_FILTER_x_OPTION` avec les options de filtrage `ACK`, `TOSMD`, `TOSMT`, `TOSMR` ou `TOSMC`, la variable `QOS_CLASS_x_DIRECTION` avec le paramètre 'down' sera alors ignorée.

QOS_FILTER_x_IP_EXTERN On indique dans cette variable les adresses-IP et les adresses de domaine du réseau externe qui doivent être filtrés (elle se rapporte à la variable `QOS_INTERNET_DEV`). Les adresses sont séparées par un espace et peuvent être combinés librement. L'ensemble fonctionne de la même manière que `QOS_FILTER_x_IP_INTERN`. Cette variable peut aussi rester vide.

Important: Si vous ajoutez la variable `QOS_FILTER_x_OPTION` avec les options de filtrages `ACK`, `TOSMD`, `TOSMT`, `TOSMR` ou `TOSMC`, la variable `QOS_CLASS_x_DIRECTION` avec le paramètre 'down' sera alors ignorée.

QOS_FILTER_x_PORT On peut paramétrer dans cette variable un port ou une plage de ports, ils seront séparés par un espace et peuvent être combinés librement. Si la variable est vide, le trafic se fait sur tous les ports.

Au sujet de la plage de port : si l'on souhaite filtrer l'ensemble des ports de 5000 à 5099, on indiquera :

```
QOS_FILTER_x_PORT='5000-5099'
```

Un autre exemple : On voudrait filtrer le trafic des ports 20 à 21, 137 à 139 et le port 80, dans la même classe. Alors, on indiquera :

```
QOS_FILTER_x_PORT='20-21 137-139 80'
```

Cette variable peut aussi être laissée vide.

Important :

- Avec le filtrage de port, la variable `QOS_FILTER_x_PORT_TYPE` doit aussi être activée.
- Si la variable `QOS_FILTER_x_OPTION` est activée avec les options de filtrages `ACK`, `TOSMD`, `TOSMT`, `TOSMR` ou `TOSMC`, la plage de ports sera ignorée.

QOS_FILTER_x_PORT_TYPE Cette variable est seulement importante, que si la variable `QOS_FILTER_x_PORT` est utilisée, dans ce cas on peut la paramétrer (autrement elle sera simplement ignorée).

Puisque les ports se différencient entre les ports du service-client et les ports du service-serveur, vous devez indiquer ici si les ports du serveur ou du client sont visés. Les ordinateurs du réseau seront considérés comme points de référence.

Les réglages suivants sont possibles :

```
QOS_FILTER_x_PORT_TYPE='client'
QOS_FILTER_x_PORT_TYPE='server'
```

Depuis la version 2.1 la combinaison des deux arguments dans une seule variable est possible, pour le trafic sur les ports dans n'autre propre réseau et pour le trafic sur les ports externe pour Internet, dans la même classe, par exemple :

```
QOS_FILTER_x_PORT_TYPE='client server'
```

Cela correspond à l'élaboration de deux filtres semblables, dans lequel on a placé le client et le serveur sur la même variable `QOS_FILTER_x_PORT_TYPE`.

QOS_FILTER_x_OPTION Avec cette variable on indique d'autres options pour le filtre actif. Il ne faut pas spécifier plus d'un paramètre, car une combinaison de plusieurs paramètres dans le même filtre n'a pas de sens. En revanche, il est parfaitement possible et même parfois utile, de filtrer les paquets-ACK, et un 2ème paquet. Par exemple filtrer les paquets-TOSMD, dans la même destination de la classe (voir QOS_FILTER_x_CLASS).

ACK Paquet de confirmation.

Si on applique ce paquet dans la variable option, lorsqu'un paquet de données arrive il sera renvoyé un paquet de confirmation comme un "accusé de réception". Si vous téléchargez de grands fichiers beaucoup de paquets sont transmis, pour chaque paquet transmis vous devez envoyer une confirmation ou un accusé de réception pour indiquer que le paquet de données est bien arrivé. Si la confirmation se fait attendre, l'expéditeur attendra celui-ci avant d'envoyer un nouveau paquet, le prochain paquet ne sera pas pour vous si la confirmation n'est pas reçue.

Toutes les connexions asymétriques sont particulièrement importantes, actuellement (le débit montant/descendant est inégal) dans la plus par des offres DSL privées. Généralement, le débit montant est beaucoup plus faible, poussé à ses limites les paquets sont empilés avant d'être envoyés et indiscutablement quelque part dans l'immense tas, il y a les petits paquets de confirmation. Dans des circonstances normales, cela fonctionne séquentiellement. Jusqu'à ce que le paquet de confirmation à son tour est envoyé, il serait bien que notre expéditeur de données puisse faire une petite pause, mais ce n'est pas bon pour le débit descendant.

Nous devons veiller à ce que les paquets de confirmations soient bien à la suite, pour que les paquets de confirmation soient envoyés "normalement" au fur et à mesure, afin que l'expéditeur reçoive bien la confirmation. Cette option est logique d'être combinée à une classe, pour cet exemple d'application.

ICMP Paquet-Ping (Protocole ICMP)

On utilise un paquet Ping pour mesurer le temps en seconde que met ce paquet pour aller du point A au point B. Si vous voulez donner au paquet Ping une priorité plus élevée, vous pouvez indiquer cette valeur dans la variable option. Cette valeur n'a rien à voir pour jouer sur Internet. Ce n'est pas parce que vous avez activé le paquet Ping que vous serez prioritaire et que vous aurez un super Ping pour jouer sur le Net...

IGMP Paquet-IGMP (Protocole IGMP)

Si vous utilisez la TV par IP, le protocole IGMP sera utile pour le filtrage et la hiérarchisation.

TCPSMALL Petit paquet TCP

Vous pouvez utiliser ce filtrage pour les requêtes HTTP(s) sortantes, ce filtre sera prioritaire. Une combinaison avec un port de destination est possible et judicieux. Taille des paquets TCP : max. 800 octets.

TCP Paquet-TCP (Protocole TCP)

Avec ce paquet on filtre uniquement les paquets qui utilisent le protocole TCP.

UDP Paquet-UDP (Protocole UDP)

Avec ce paquet on filtre uniquement les paquets qui utilisent le protocole UDP.

TOS* Type of Service

TOS "Type of Service" est une application qui place 4 Bit-TOS dans l'entête IP pour chaque paquet envoyé. Ceux-ci ont pour effet de modifier la manière dont les paquets sont traités. Par exemple, on peut placer TOS-Minimum-Delay pour Améliore la réactivité SSH et TOS-Maximum-Troughput pour expédition de fichiers. Généralement ce sont des programmes Linux/Unix qui utilisent plus fréquemment ces Bit-TOS, que les programme Windows. En outre, on peut placer ces Bit-TOS pour certains paquets à destination des IT Firewall. Cela dépend bien sûr que les Routeurs acceptent ou pas les Bit-TOS. En réalité pour fl4l Minimum-Delay et Maximum-Throughput présentent un vrai intérêt.

TOSMD - TOS Minimum-Delay Ce service est utilisé pour améliorer la réactivité des connexions en réduisant le délai de transmission des paquets, Bit-TOS est recommandé pour le SSH, Telnet, FTP (contrôle), TFTP.

TOSMT - TOS Maximum-Troughput Ce service est utilisé pour améliorer le débit des gros de fichier, au prix d'une possible détérioration de l'interactivité de la session. Les temps de latence ne sont pas importants, Bit-TOS est recommandé pour le FTP-data et WWW.

TOSMR - TOS Maximum-Reliability Ce service est utilisé pour avoir la certitude que les données arrivent sans perte, améliore la fiabilité, on évite un revoie de paquet inutile, Bit-TOS est recommandé pour SNMP et DNS

TOSMC - TOS Minimum-Cost Ce service est utilisé pour minimiser le délai, une meilleure rentabilité, Bit-TOS est recommandé pour NNTP, SMTP et ICMP.

DSCP* Differentiated Services Code Point

Le DSCP est appelé marquage conformément défini dans la RFC 2474. Ce processus de marquage a largement remplacé le TOS depuis 1998.

Le filtrage des classes DSCP peut être configuré comme ceci :

```
QOS_FILTER_x_OPTION='DSCPef'  
QOS_FILTER_x_OPTION='DSCPcs3'
```

S'il vous plaît, faite attention que le DSCP soit écrit en lettre capital et la classe en minuscule.

Les classes suivantes peuvent être utilisées :

af11-af13, af21-af23, af31-af33, af41-af43, cs1-cs7, ef et be (par défaut)

1.1.2. Applications et exemples

Comment doit on configurer OPT_QoS précisément ? voici quelques exemples concret :

- Exemple 1 : exemple simple pour le partager de la bande passante sur trois ordinateurs.
- Exemple 2 : exemple de configuration pour la distribution de la bande passante sur 2 ordinateurs et en plus une seconde répartition de la bande passante sur les ports des ordinateurs respectif, en plus il restera de la B-P.
- Exemple 3 : exemple de fonctionnement général de QoS avec un peut plus de détail.
- Exemple 4 : exemple de configuration des paquets-ACK avant d'être transmis, sur le débit descendant de sorte que le débit montant ne chute pas.

Exemple 1

L'objectif de cet exemple simple, est de distribuer la bande passante sur 3 ordinateurs.

Pour ce faire, nous allons créer 4 classes (voir QOS_CLASS_N et se qui suit) pour différentes vitesses (voir QOS_CLASS_x_MINBANDWIDTH / QOS_CLASS_x_MAXBANDWIDTH) cela dépend aussi de la classe 0 (voir QOS_CLASS_x_PARENT) Donc directement de l'interface pour "up" ou "down" (voir QOS_CLASS_x_DIRECTION).

La classe 4 est éventuellement pour un hôte en plus, avec moins de band passante. nous indiquons dans QOS_INTERNET_DEFAULT_DOWN='4' pour tous le transport non filtré et une quatrième Classe pour les "invités". Étant donné que nous avons rarement d'hôte, la bande passante sera la même pour les 3 autres classes, chaque ordinateur recevra 1/3 de l'ensemble de la bande passante, c'est-à-dire approximativement 256Kibit/s.

Nous avons tout d'abord configurer la structure fondamentale. Maintenant, nous devons encore choisir pour le réglage quelle transport pour quelle classe.

Pour ce faire, nous allons utiliser des filtres, pour associer le trafic à chacune des classes. Nous allons créer ainsi des filtres pour les 3 ordinateurs (voir QOS_FILTER_N et se qui suit) et la classification des filtres pour chaque classe (voir QOS_FILTER_x_CLASS). Maintenant, nous pouvons avec QOS_FILTER_x_IP_INTERN, QOS_FILTER_x_IP_EXTERN, QOS_FILTER_x_PORT, QOS_FILTER_x_PORT et QOS_FILTER_x_OPTION déterminer le réglage de chacune des classes, pour laquelle appartient le filtre.

Vous pouvez voir le principe dans la figure 1.1 l'Interface 0 les 3 classe 1, 2 et 3, les 3 filtres F1, F2 et F3.

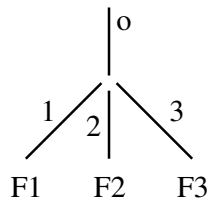


FIGURE 1.1.: exemple 1

La configuration ressemble alors à :

Trois ordinateurs filtrés par IP reçoivent chacune 1/3 du débit, si il n'y a pas d'hôte en plus

```

OPT_QOS='yes'
QOS_INTERNET_DEV_N='1'
QOS_INTERNET_DEV_1='ppp0'
QOS_INTERNET_BAND_DOWN='768Kibit/s'
QOS_INTERNET_BAND_UP='128Kibit/s'
QOS_INTERNET_DEFAULT_DOWN='4'
QOS_INTERNET_DEFAULT_UP='0'

QOS_CLASS_N='4'

QOS_CLASS_1_PARENT='0'
QOS_CLASS_1_MINBANDWIDTH='232Kibit/s'
QOS_CLASS_1_MAXBANDWIDTH='768Kibit/s'
QOS_CLASS_1_DIRECTION='down'
QOS_CLASS_1_PRIO=''

```

1. Documentation du paquetage QOS

```
QOS_CLASS_2_PARENT='0'  
QOS_CLASS_2_MINBANDWIDTH='232Kibit/s'  
QOS_CLASS_2_MAXBANDWIDTH='768Kibit/s'  
QOS_CLASS_2_DIRECTION='down'  
QOS_CLASS_2_PRIO=''
```

```
QOS_CLASS_3_PARENT='0'  
QOS_CLASS_3_MINBANDWIDTH='232Kibit/s'  
QOS_CLASS_3_MAXBANDWIDTH='768Kibit/s'  
QOS_CLASS_3_DIRECTION='down'  
QOS_CLASS_3_PRIO=''
```

```
QOS_CLASS_4_PARENT='0'  
QOS_CLASS_4_MINBANDWIDTH='72Kibit/s'  
QOS_CLASS_4_MAXBANDWIDTH='768Kibit/s'  
QOS_CLASS_4_DIRECTION='down'  
QOS_CLASS_4_PRIO=''
```

```
QOS_FILTER_N='3'
```

```
QOS_FILTER_1_CLASS='1'  
QOS_FILTER_1_IP_INTERN='192.168.0.2'  
QOS_FILTER_1_IP_EXTERN=''  
QOS_FILTER_1_PORT=''  
QOS_FILTER_1_PORT_TYPE=''  
QOS_FILTER_1_OPTION=''
```

```
QOS_FILTER_2_CLASS='2'  
QOS_FILTER_2_IP_INTERN='192.168.0.3'  
QOS_FILTER_2_IP_EXTERN=''  
QOS_FILTER_2_PORT=''  
QOS_FILTER_2_PORT_TYPE=''  
QOS_FILTER_2_OPTION=''
```

```
QOS_FILTER_3_CLASS='3'  
QOS_FILTER_3_IP_INTERN='192.168.0.4'  
QOS_FILTER_3_IP_EXTERN=''  
QOS_FILTER_3_PORT=''  
QOS_FILTER_3_PORT_TYPE=''  
QOS_FILTER_3_OPTION=''
```

L'option QOS_INTERNET_DEFAULT_UP a été mis sur 0 parce le débit montant ne devrait pas être limité.

Exemple 2

L'objectif de cet exemple est de distribuer la bande passante sur 2 ordinateurs, une seconde répartition de la bande passante se fera sur les ports des deux ordinateurs respectif, il restera de la Bande passante pour le transfert.

Pour cela, nous avons de 2 classes avec leur vitesse respective, ils dépendent directement de l'interface pour "up" et/ou "down" (voir le premier exemple). Ensuite nous allons créer pour le

première ordinateur de la première classe deux autre sous classes. Les deux sous classes sont créées comme la première classe directement sur l'interface, avec toutefois une particularité : la variable `QOS_CLASS_x_PARENT` n'est pas sur 0, mais sur le nombre de classe à laquelle les sous-classes sont attachées. Par exemple pour `QOS_CLASS_1`, nous devons maintenant ajouté 1 dans la classes `QOS_CLASS_1`, suivante. Il en sera de même pour le second ordinateur. On attache à nouveau deux sous classes à la classe du deuxième ordinateur, ils doivent être sur 2. Ceci peut être fait, non seulement pour deux ordinateurs, mais pour autant d'ordinateur que vous voulez. De même on peut créer autant de sous-classes par rapport à la classe supérieur.

nous avons tout d'abord configurer la structure fondamentale. Ensuite nous devons assigner les filtres de chaque classe pour le trafic. (Voir le premier exemple)

Nous allons créer 2 filtres pour le premier ordinateur et 2 filtres pour le deuxième ordinateur. Il y aura respectivement un filtre pour le port, et d'un filtre pour le transfert des donnés. Il faut absolument tenir compte des séquences. le premier filtre uniquement pour le port, puis le reste. Il est impératif de respecter l'ordre. Autrement le filtre sera déjà attribué pour le reste les classes.

Vous pouvez voir le principe dans la figure 1.2 l'Interface 0, les 6 classes 1, 2, 3, 4, 5, et 6, les 4 filtres F1, F2, F3 et F4.

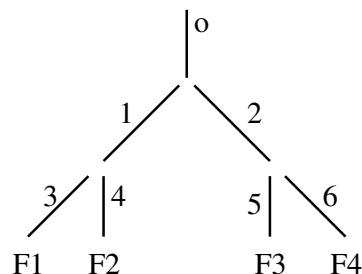


FIGURE 1.2.: exemple 2

La configuration ressemble alors à :

Deux classes pour 2 ordinateurs qui obtiennent 1/2 de BP, deux classes pour les ports qui obtiennent 2/3 de la B-P, il reste donc 1/3 de B-D pour chaque classe parent :

```

OPT_QOS='yes'
QOS_INTERNET_DEV_N='1'
QOS_INTERNET_DEV_1='ppp0'
QOS_INTERNET_BAND_DOWN='768Kibit/s'
QOS_INTERNET_BAND_UP='128Kibit/s'
QOS_INTERNET_DEFAULT_DOWN='7'
QOS_INTERNET_DEFAULT_UP='0'

QOS_CLASS_N='6'

QOS_CLASS_1_PARENT='0'
QOS_CLASS_1_MINBANDWIDTH='384Kibit/s'
QOS_CLASS_1_MAXBANDWIDTH='768Kibit/s'
QOS_CLASS_1_DIRECTION='down'
QOS_CLASS_1_PRIO=' '

QOS_CLASS_2_PARENT='0'
  
```

1. Documentation du paquetage QOS

```
QOS_CLASS_2_MINBANDWIDTH='384Kibit/s'  
QOS_CLASS_2_MAXBANDWIDTH='768Kibit/s'  
QOS_CLASS_2_DIRECTION='down'  
QOS_CLASS_2_PRIO=''
```

```
QOS_CLASS_3_PARENT='1'  
QOS_CLASS_3_MINBANDWIDTH='256Kibit/s'  
QOS_CLASS_3_MAXBANDWIDTH='768Kibit/s'  
QOS_CLASS_3_DIRECTION='down'  
QOS_CLASS_3_PRIO=''
```

```
QOS_CLASS_4_PARENT='1'  
QOS_CLASS_4_MINBANDWIDTH='128Kibit/s'  
QOS_CLASS_4_MAXBANDWIDTH='768Kibit/s'  
QOS_CLASS_4_DIRECTION='down'  
QOS_CLASS_4_PRIO=''
```

```
QOS_CLASS_5_PARENT='2'  
QOS_CLASS_5_MINBANDWIDTH='256Kibit/s'  
QOS_CLASS_5_MAXBANDWIDTH='768Kibit/s'  
QOS_CLASS_5_DIRECTION='down'  
QOS_CLASS_5_PRIO=''
```

```
QOS_CLASS_6_PARENT='2'  
QOS_CLASS_6_MINBANDWIDTH='128Kibit/s'  
QOS_CLASS_6_MAXBANDWIDTH='768Kibit/s'  
QOS_CLASS_6_DIRECTION='down'  
QOS_CLASS_6_PRIO=''
```

```
QOS_FILTER_N='4'
```

```
QOS_FILTER_1_CLASS='3'  
QOS_FILTER_1_IP_INTERN='192.168.0.2'  
QOS_FILTER_1_IP_EXTERN=''  
QOS_FILTER_1_PORT='80'  
QOS_FILTER_1_PORT_TYPE='client'  
QOS_FILTER_1_OPTION=''
```

```
QOS_FILTER_2_CLASS='4'  
QOS_FILTER_2_IP_INTERN='192.168.0.2'  
QOS_FILTER_2_IP_EXTERN=''  
QOS_FILTER_2_PORT=''  
QOS_FILTER_2_PORT_TYPE=''  
QOS_FILTER_2_OPTION=''
```

```
QOS_FILTER_3_CLASS='5'  
QOS_FILTER_3_IP_INTERN='192.168.0.3'  
QOS_FILTER_3_IP_EXTERN=''  
QOS_FILTER_3_PORT='80'  
QOS_FILTER_3_PORT_TYPE='client'  
QOS_FILTER_3_OPTION=''
```

```
QOS_FILTER_4_CLASS='6'
```

```
QOS_FILTER_4_IP_INTERN='192.168.0.3'
QOS_FILTER_4_IP_EXTERN=''
QOS_FILTER_4_PORT=''
QOS_FILTER_4_PORT_TYPE=''
QOS_FILTER_4_OPTION=''
```

Avec cet exemple l'option `QOS_INTERNET_DEFAULT_DOWN` a été choisie de telle sorte que le transfert qui n'est pas assigné par une classe et un filtre, soit mis dans une classe non existante (non paramétrée). La raison, dans cet exemple, c'est qu'il reste 1/3 de bande passante non affectée. c'est pour cela que l'on conduit cette B-P sur une classe non existante, en plus elle est transmise très lentement. S'il reste de la B-P dans la configuration vous devez absolument, l'indiquer dans une classe existante.

L'option `QOS_INTERNET_DEFAULT_UP` a été réglé sur 0 pour que Upstream (ou débit montant) ne soit pas limité.

Exemple 3

Exemple du mode de fonctionnement de QoS en général ou presque.

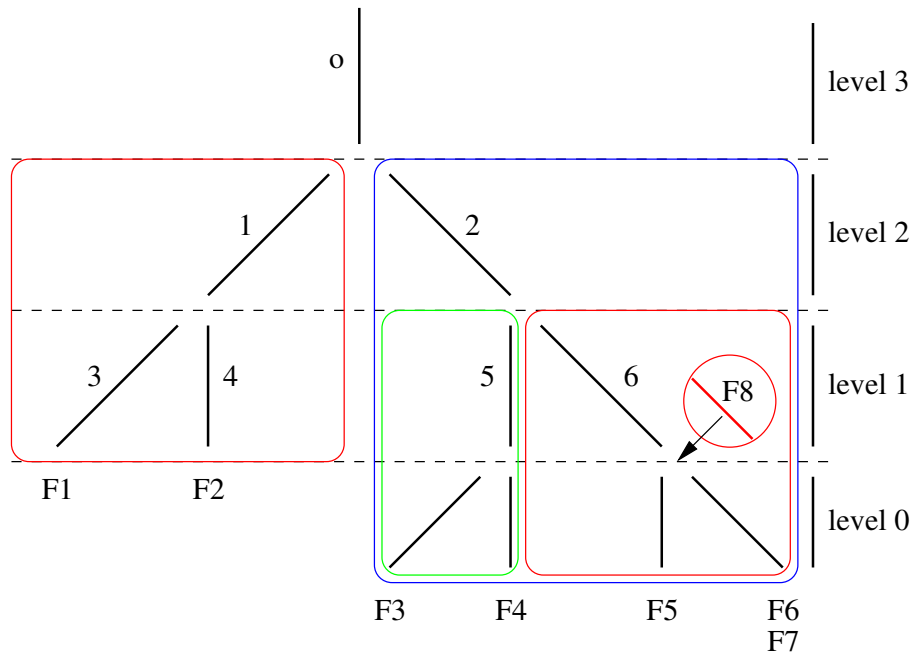


FIGURE 1.3.: exemple 3

Dans la figure 1.3 nous allons revoir la répartition du deuxième exemple, mais avec une extension supplémentaire. On ajoutées deux sous classe à la sous classe du deuxième niveau. Il est possible d'imbriquer des classes encore plus profondément dans cette représentation, la limite actuelle se situe ici à 8 branches, on peut produire au maximum de 7 branches après l'interface 0, ensuite c'est terminé. Cependant, dans la "largeur" aucune limite n'est fixée. Donc vous pouvez rajouter à l'une des sous-classes autant de classes que vous voulez.

On peut voir sur cette figure, qu'il est aussi possible d'intégrer plus d'un filtre à une classe comme avec la classe 10. Il est à noter au sujet du filtrage, qu'il n'est pas possible de placer un filtre au milieu de "l'arborescence" comme indiqué en F8.

Voyons maintenant, encore une fois le sens des classes et des sous-classes. Les classes sont réglementées et sont utilisées pour régler la vitesse de connexion. La répartition de la vitesse est effectué avec la variable `QOS_CLASS_x_MINBANDWIDTH`. Cependant, cela peut avoir un inconvénient si par exemple, toutes les sous classes dépendent d'une classe. Si on donnait, par exemple, à un ordinateur la moitié de la bande passante et l'autre moitié répartie en 2/3 pour le http et 1/3 pour le reste, c'est-à-dire 2/6 et 1/6 de l'ensemble. Cela se présente maintenant : à pleine charge chacune des branches reçoit une moitié. Quant à la deuxième, oui il ne reste pour le http que 2/6. Cependant, le 2e ordinateur ne reçoit pas les 2/6, mais cela est réparti selon la méthode décrite. Pour éviter ceci, on crée des sous-classes. Le trafic des classes est distribué d'abord aux sous-classes, si celle-ci ne demande pas le transfert complet de la bande passante, le restant sera réparti sur d'autres classes. dans la figure, les zones qui sont ensembles sont entourées, en Rouge = 1, bleu = 2, vert = 5 et orange = 6

Exemple 4

Exemple de configuration, pour la priorité des paquets-ACK et pour que le Downstream (ou débit descendant) ne chute pas, il nous faut simultanément un Upstream (ou débit montant) fort :

```
OPT_QOS='yes'
QOS_INTERNET_DEV_N='1'
QOS_INTERNET_DEV_1='ppp0'
QOS_INTERNET_BAND_DOWN='768Kibit/s'
QOS_INTERNET_BAND_UP='128Kibit/s'
QOS_INTERNET_DEFAULT_DOWN='0'
QOS_INTERNET_DEFAULT_UP='2'
```

Nous configurons ici l'interface pour l'accès Internet (DSL) avec le protocole ppp0 et nous indiquons le débit de la bande passante Up/Down (ou montant/descendant). Qui est donné par (quelques fournisseur d'accès). Il est nécessaire de réduire au minimum la quantité du débit montant de la bande passante en Kibibit, pour cela vous devez faire des tests.

Étant donné que nous ne voulons pas de classe avec Downstream (ou débit descendant) nous indiquons

```
QOS_INTERNET_DEFAULT_DOWN='0'
```

Nous indiquons 2 pour le nombre de classe standard avec Upstream (ou débit montant). L'interface réseau affecté eth0 a un débit de 10Mibit/s

```
QOS_CLASS_N='2'

QOS_CLASS_1_PARENT='0'
QOS_CLASS_1_MINBANDWIDTH='127Kibit/s'
QOS_CLASS_1_MAXBANDWIDTH='128Kibit/s'
QOS_CLASS_1_DIRECTION='up'
QOS_CLASS_1_PRIO=''
```


1. Documentation du paquetage QOS

Il s'agit ci-dessous de construire la classe dans laquelle nous allons insérer les paquets-ACK pour les (accusés de réception). Les paquets-ACK sont assez petites, c'est pour cette raison qu'il on besoin de peu de bande passante. Néanmoins, nous voulons en aucune façon partager ses 127Kibit. Nous laisserons donc 1Kibit/s pour le reste.

```
QOS_CLASS_2_PARENT='0'
QOS_CLASS_2_MINBANDWIDTH='1Kibit/s'
QOS_CLASS_2_MAXBANDWIDTH='128Kibit/s'
QOS_CLASS_2_DIRECTION='up'
QOS_CLASS_2_PRIO=' '
```

Dans la classe ci-dessus, on insère le reste (tout sauf les paquets-ACK). La bande passante que nous allons indiquer dans cette classe est 1Kibit/s restants, donc (128-127=1). La somme de 1Kibit/s que nous avons enregistré, n'est pas limité.

```
QOS_CLASS_2_MAXBANDWIDTH='128Kibit/s'
```

Notre première classe utilisera peut-être, à peine toute la bande passante attribuée et si il en reste un peu, alors, le reste de la B-P, passera sur la deuxième classe. Si l'on veut diviser encore davantage le Upstream (ou débit montant) (ce qui est généralement le cas), toutes les autres sous-classes "dépendent" de cette classe. Il faut, bien sûr paramétrer la variable `QOS-INTERNET_DEFAULT_UP` en conséquence.

```
QOS_FILTER_N='1'

QOS_FILTER_1_CLASS='1'
QOS_FILTER_1_IP_INTERN=''
QOS_FILTER_1_IP_EXTERN=''
QOS_FILTER_1_PORT=''
QOS_FILTER_1_PORT_TYPE=''
QOS_FILTER_1_OPTION='ACK'
```

Ce filtre, filtre tous les paquets qui s'appliquent au option de filtrage, donc les paquets-ACK. Grâce à l'enregistrement de la variable `QOS_FILTER_1_CLASS='1'` nous somme sur de filtrer les paquets de la 1ère classe.

Pour tester, on doit rechercher au mieux une ou plusieurs bonnes sources en "envoyant/recevant des données", en sachant que l'on doit utiliser tout le débit montant/descendant et de faire "chauffer les câbles". Il faut jeter un coup d'oeil sur l'état du trafic avec Imonc (s'il est installé). Le meilleur moyen est de le faire sans activer QoS.

Le Downstream (ou débit descendant) ne devrait pas du tout chuter ou beaucoup moins fort que sans cette configuration. Comme je l'ai dit, on peut encore améliorer la situation du débit montant de la bande passante par incrémentation en Kibibit pour réduire au minimum et de vérifier les effets. Chez moi, par exemple, le meilleur débit atteint est de 121Kibit/s (en plus pas de chute avec le débit descendant). Il faut bien sûr pour chaque classe prendre en considération les valeurs de MAXBANDWIDTH et MINBANDWIDTH.

A. Annexe du paquetage QOS

Table des figures

1.1. exemple 1	11
1.2. exemple 2	13
1.3. exemple 3	15

Liste des tableaux

Index

OPT_QOS, [3](#)

QOS_CLASS_N, [4](#)

QOS_CLASS_x_DIRECTION, [5](#)

QOS_CLASS_x_LABEL, [6](#)

QOS_CLASS_x_MAXBANDWIDTH, [5](#)

QOS_CLASS_x_MINBANDWIDTH, [5](#)

QOS_CLASS_x_PARENT, [4](#)

QOS_CLASS_x_PRIO, [6](#)

QOS_FILTER_N, [6](#)

QOS_FILTER_x_CLASS, [7](#)

QOS_FILTER_x_IP_EXTERN, [8](#)

QOS_FILTER_x_IP_INTERN, [7](#)

QOS_FILTER_x_OPTION, [8](#)

QOS_FILTER_x_PORT, [8](#)

QOS_FILTER_x_PORT_TYPE, [8](#)

QOS_INTERNET_BAND_DOWN, [3](#)

QOS_INTERNET_BAND_UP, [4](#)

QOS_INTERNET_DEFAULT_DOWN, [4](#)

QOS_INTERNET_DEFAULT_UP, [4](#)

QOS_INTERNET_DEV_N, [3](#)

QOS_INTERNET_DEV_x, [3](#)