netfilter/iptables FAQ

Harald Welte <laforge@gnumonks.org>
Traduit par Fabrice MARIE*fabrice@netfilter.org* Version $Revision: 564 $, $Date: 2003-11-26 17:33:57 +0100 (mié, 26 nov 2003) $, Dernière traduction le Jeudi 08 Août 2002

Ce document contient les réponses aux questions posées le plus souvent sur la liste de diffusion. Les commentaires/additions/clarifications seront appréciés et doivent être dirigés vers le mainteneur de cette FAQ.

Table des matières

1 Questions Générales

Cette section couvre les questions générales sur Netfilter (ou pas) que nous avons rencontrées sur la liste.

1.1 Où puis-je récupérer netfilter/iptables ?

Netfilter et IPtables sont intégrés dans les noyaux de la série 2.4.x. Veuillez télécharger un noyau récent à partir de

`http://www.kernel.org/` ou l'un de ses miroirs.

L'outil en userspace 'iptables' est disponible sur l'un des sites de Netfilter :

`http://www.netfilter.org/` ,

`http://www.iptables.org/` ,

`http://netfilter.samba.org/` ,

`http://netfilter.gnumonks.org/` or

`http://netfilter.filewatcher.org/` .

1.2 Existe-t'il un portage de Netfilter pour les Linux 2.2 ?

Non, il n'y en a pas pour le moment. Mais si quelqu'un veut commencer, ça ne devrait pas être trop dur grâce à l'interface propre de la pile de protocoles réseaux.

Dites nous si vous commencez à travailler la dessus.

1.3 Y a-t'il un module Helper pour le suivi de connexions/NAT ?

Si vous êtes habitués au masquerading sous Linux 2.2, vous avez toujours utilisé le module 'ip_masq_icq' pour que les connexions client-à-client fonctionnent avec ICQ.

Personne n'a ré-implementé ce module pour Netfilter, parce que le protocole ICQ n'est pas beau :) Mais je pense que c'est juste une question de temps avant qu'un tel module soit disponible.

Rusty a dit une fois que seuls les protocoles, disposant d'au moins un client et un serveur libres, seront intégrés dans la distribution principale de netfilter. Pour ce qui est de ICQ, il y a seulement des clients libres, et donc il ne correspond pas aux critères. ('free' comme dans liberté, et pas comme bière gratuite), c'est-à-dire la définition de Stallmann).

1.4 Où sont passés les modules ip_masq_vdolive / ip_masq_quake / ...

Quelques-uns d'entre eux ne sont pas nécessaires, et quelques-uns n'ont pas été encore portés pour netfilter. Netfilter implémente un suivi de connexions complet, y compris en UDP, et a une politique visant à modifier les paquets le moins possible, et donc, de temps en temps, ça marche tout simplement.

1.5 Qu'est-ce que 'patch-o-matic' et comment s'en sert-t'on ?

Les noyaux 2.4.x sont des versions stables, donc on ne peut pas soumettre nos versions de développement directement dans la branche principale. Tout notre code est développé et testé dans patch-o-matic d'abord. Si vous voulez utiliser une des fonctionnalités dernier cri, vous aurez à appliquer un ou plusieurs patches à partir de patch-o-matic. Vous trouverez patch-o-matic dans le dernier package iptables (ou biensûr dans le CVS), à télécharger à partir du site netfilter.

Maintenant patch-o-matic a trois différentes options :
– make pending-patches
– make most-of-pom
– make patch-o-matic
La première option est juste là pour s'assurer que tous les patches les plus importants, qui ont déjà été envoyés aux mainteneurs du noyau de toutes façons, sont appliqués à votre noyau.

La seconde, 'most-of-pom', vous propose aussi chaque nouvelle fonctionnalité qui peux être appliquée sans conflit.

La troisième option, 'patch-o-matic' est pour les experts qui veulent avoir accès à tous les patches, mais faites attention, il y aura certainement des conflits dans ce mode.

patch-o-matic a une interface utilisateur agréable. Entrez simplement :

```
make most-of-pom
```

ou, si les sources de votre noyau ne sont pas dans `/usr/src/linux`, alors utilisez :

```
make KERNEL_DIR={votre_repertoire_noyau} most-of-pom
```

dans le répertoire principal du package iptables. Pour chaque patch, patch-o-matic vérifie que celui-ci pourrait s'appliquer sans erreur. Si le patch peut s'appliquer, alors vous verrez apparaître un petit prompt où vous pourrez demander plus d'informations, appliquer le patch, passer au suivant, etc...

Pour plus d'informations à propos de patch-o-matic, allez voir le Netfilter Extensions HOWTO, que vous pourrez trouver sur

http://www.netfilter.org/documentation/index.html#HOWTO

1.6 Où puis-je trouver 'ipnatctl' et plus d'informations à propos de ça ?

'ipnatctl' était utilisé pour configurer les règles de NAT à partir du userspace dans les révisions de développement au début des noyaux 2.3.x. On en n'a plus besoin et il n'est donc plus disponible. Toutes ses fonctions sont maintenant remplies par netfilter. Allez regarder le 'NAT HOWTO' sur le site de Netfilter.

2 Problèmes lors de la compilation

2.1 Je ne peux pas compiler iptables-1.1.1 avec les noyaux >= 2.4.0-test4.

C'est un problème connu. Le mécanisme, pour détecter quels patchs doivent être appliqués, est cassé. Essayez d'utiliser "make build" à la place de "make".

Une meilleure solution : mettez à jour avec iptables-1.1.2 ou ultérieur.

2.2 Je ne peux pas compiler iptables-1.1.9 avec les noyaux récents (>= 2.3.99-pre8)

Les structures internes de iptables ont changé. Mettez à jour avec iptables > 1.1.1.

2.3 Quelques patches de patch-o-matic de iptables >= 1.2.1a ne compilent plus avec un noyau >= 2.4.4

Veuillez utiliser une version récente d'iptables.

2.4 ipt_BALANCE, ip_nat_ftp, ip_nat_irc, ipt_SAME et ipt_NETMAP ne compilent pas

Vous avez probablement des problèmes de compilation avec une fonction qui s'appelle `ip_nat_setup_info`.

Si vous utilisez iptables <= 1.2.2, vous **DEVEZ** appliquer les patches 'dropped-table' et 'ftp-fixes'.

Si vous utilisez iptables > 1.2.2, ou une version CVS récente, **n'appliquez PAS** le patch 'dropped-table', puisqu'il est incompatible avec BALANCE, NETMAP, irc-nat, SAME et talk-nat.

2.5 J'utilise la série de noyaux d'Alan Cox, les 2.4.x-acXXX, et j'ai des problèmes.

L'équipe de développement de netfilter base son travail sur les noyaux de Linus, donc si vous voulez utiliser la série -ac, faites-le à vos propres risques.

2.6 ERROR : Invalid option KERNEL_DIR=/usr/src/linux-2.4.19

Je parie que vous essayez de lancer quelque chose comme ça :

```
# ./runme pending KERNEL_DIR=/usr/src/linux-2.4.19
```

Mais bash/sh n'est pas comme make, et on ne peut pas passer de valeur de variable en parametre. Vous devez assigner la variable avant de lancer le script runme :

```
# KERNEL_DIR=/usr/src/linux-2.4.19 ./runme pending
```

3 Problèmes au lancement.

3.1 'NAT : X dropping untracked packet Y Z aaa.aaa.aaa.aaa -> 224.bbb.bbb.bbb'

Ce message est affiché par le code de NAT parce que des paquets multicast atteignent la table de NAT, et que le code de suivi de connexion ne gère pas actuellement les paquets multicast. Si vous n'avez aucune idée de ce qu'est le multicast ou si vous n'en avez pas besoin, utilisez :

```
iptables -t mangle -I PREROUTING -j DROP -d 224.0.0.0/8
```

3.2 NAT : X dropping untracked packet Y Z aaa.aaa.aaa.aaa -> bbb.bbb.bbb.bbb

Syslog ou ma console affiche le message :

```
NAT: X dropping untracked packet Y Z aaa.aaa.aaa.aaa -> bbb.bbb.bbb.bbb
```

Ce message est affiché par le code de NAT. Il détruit le paquet, parce que pour pouvoir mener à bien la NAT, NAT a besoin d'avoir un paquet ayant des infos valides de suivi de connexions. Ce message est affiché pour chaque paquet dont le code de suivi de connexions ne permet pas de déterminer les informations de suivi.

Les raisons possibles sont :
– Nombre maximum d'entrées dans la base de données de suivi de connexions a été atteint,
– Ne pouvait pas déterminer le n-uplet inverse (multicast, broadcast),
– 'kmem_cache_alloc' échoue (pas assez de mémoire),
– réponse à une connexion non confirmée.,
– paquet multicast (voir la question suivante),
– paquet icmp trop petit,
– paquet icmp fragmente,
– 'checksum' du paquet icmp erronée.

Si vous voulez avoir plus de détails pendant le log de ces paquets (si vous suspectez que ce sont des paquets de scanners), utilisez la règle suivante :

```
iptables -t mangle -A PREROUTING -j LOG -m state --state INVALID
```

Et oui ! Vous devez mettre cette règle dans la table 'mangle', parce que le paquet sera détruit par le code de NAT avant qu'il ne parvienne dans la table 'filter'.

3.3 Je n'arrive pas à utiliser netfilter avec le code de Bridging.

Donc vous voulez construire un firewall totalement transparent ? Bonne idée ! A partir du noyau 2.4.16, vous avez toujours besoin de patcher votre noyau pour que ça marche. Vous le trouverez sur la page web du projet *suivant* http://bridge.sourceforge.net/ .

3.4 Le module IRC est incapable de gérer le DCC RESUME.

Et bien... c'est à moitié vrai. Seul le module de NAT est incapable de les gérer. Si vous utilisez le firewall seul, donc sans NAT, ça devrait marcher impeccablement.

3.5 Comment fonctionne SNAT vers plusieurs adresses ?

Netfilter essaye de modifier les paquets aussi peu que possible. Donc si vous avez une machine fraîchement redémarrée, et que quelqu'un derrière la NAT ouvre une connexion avec un port local 1234, la partie netfilter modifie seulement l'adresse IP et ne touche pas au numéro de port.

Aussitôt que quelqu'un d'autre ouvre une autre connexion avec le même port source, netfilter devra modifier l'adresse IP et le numéro de port si il y a seulement une IP pour la SNAT.

Mais s'il y en plus d'une disponible, alors encore une fois netfilter ne changera que l'adresse IP.

3.6 ip_conntrack : maximum limit of XXX entries exceeded

Si vous voyez ce message d'erreur dans votre syslog, ça veut dire que le code de suivi de connexions ne dispose plus d'assez de place pour ajouter une entrée. Le code de suivi de connexions gère par défaut jusqu'à 4096 connexions simultanées en général, mais cela dépend de la RAM que vous avez. Pour augmenter cette limite, tapez par exemple :

```
echo "8192" > /proc/sys/net/ipv4/ip_conntrack_max
```

Bien sur, vous pouvez utiliser n'importe quel nombre qui rentre dans un 'int' sur votre hardware (c'est à dire 32 bits pour la plupart des plates-formes). Veuillez noter que chaque connexion suivie peut avoir besoin d'un certain montant de mémoire noyau non-swappable (500 octets par connexion, pour vous donnez un ordre d'idée).

3.7 Comment puis-je lister toutes les connexions suivies/masqueradées, l'équivalent de 'ipchains -L -M', qu'on utilisait dans les 2.2 ?

Il y à un fichier dans le système de fichiers '/proc' qui s'appelle `/proc/net/ip_conntrack`. Et voila :

```
cat /proc/net/ip_conntrack
```

3.8 Comment puis-je lister toutes les tables disponibles ?

Toutes les tables disponibles sont listées avec la commande :

```
cat /proc/net/ip_tables_names
```

3.9 iptables-save / iptables-restore de iptables-1.2 génère un segfault.

Bug connu. Veuillez mettre à jour avec le dernier CVS ou utilisez iptables >= 1.2.1

3.10 'iptables -L' prend beaucoup de temps pour afficher les règles.

C'est parce que iptables résout les adresses IP avec le DNS, et fait une requête au DNS pour chaque adresse. Comme chaque règle consiste en deux adresses IP, le pire des cas revient à deux requêtes par règle.

Le problème est que, si vous utilisez des adresses IP privées (comme 10.x.x.x ou 192.168.x.x), DNS est incapable de résoudre le nom d'hôte et va faire un timeout. La somme de tous ces timeouts peut être **extrêmement** long, selon votre ensemble de règles.

Utilisez l'option '-n' (numérique) avec iptables pour éviter qu'il ne tente de résoudre les adresses IP.

3.11 Comment faire pour que la target LOG arrête d'envoyer vers la console ?

Vous devez configurer votre syslogd/klogd correctement : La target LOG enregistre en utilisant le canal 'kernel' à la priorité 'warning' (4). Voir la page man de syslogd.conf pour en apprendre plus sur les canaux et les priorités.

Par défaut, tous les messages du noyau, ayant une sévérité supérieure à 'debug' (7), sont envoyés sur la console. Si vous le montez à 4, à la place de 7, ça fera que les messages de la target LOG ne seront plus affichés sur la console.

Gardez à l'esprit que ça peut aussi empêcher d'autres messages importants d'apparaître sur la console (ça n'affecte pas syslog).

3.12 Comment construire un proxy transparent avec squid et iptables ?

D'abord, bien sûr, vous avez besoin d'une règle DNAT ou REDIRECT qui fonctionne. Utilisez REDIRECT seulement si squid tourne sur la même machine que la machine qui fait NAT. Exemple :

```
iptables -t nat -A PREROUTING -p tcp --dport 80 -j DNAT --to 192.168.22.33:3128
```

Après ça, vous devez configurer squid correctement. Nous ne pouvons donner que des exemples courts ici, allez voir la doc de squid pour plus d'infos.

Le fichier squid.conf pour squid 2.3 a besoin de quelque chose comme ça :

```
http_port 3128
httpd_accel_host virtual
httpd_accel_port 80
httpd_accel_with_proxy on
httpd_accel_uses_host_header on
```

Squid 2.4 a besoin d'une ligne **additionnelle** :

```
httpd_accel_single_host off
```

3.13 Comment dois-je utiliser la cible LOG / comment LOGuer et DROPper ?

La target LOG est ce qu'on appelle une target "non terminale", en effet, elle ne stoppe pas la traversée des paquets. Si vous utilisez la target LOG, le paquet sera enregistré, et la traversée des règles continuera à la règle suivante.

Comment dois-je faire pour LOGuer et DROPper en même temps ? Rien de plus simple, créez une chaîne custom avec deux règles :

```
iptables -N logdrop
iptables -A logdrop -j LOG
iptables -A logdrop -j DROP
```

Maintenant, à chaque fois que vous voulez LOGuer et DROPper un paquet, vous pouvez utiliser simplement "-j logdrop".

3.14 Comment faire pour arrêter le vers ('worm') XYX avec netfilter ?

La réponse courte est que vous ne pouvez pas faire ça correctement avec netfilter. La plupart des vers utilisent des protocoles de haut niveau tout à fait légitimes (ex : HTTP, SMTP (ex : un script VB attaché au courrier), ou n'importe quel exploit d'une vulnérabilité qui se trouve dans un démon qui gère un protocole de haut niveau). Par protocole de haut niveau, on entend ici un protocole qui se trouve au dessus de TCP ou IP. Comme iptables ne comprend pas ces protocoles, il est presque impossible de filtrer une partie de ces protocoles. Pour ça, vous devez utiliser un proxy filtrant.

N'utilisez pas la target string disponible dans patch-o-matic à la place d'un proxy filtrant. Ça serait attaquable en permanence avec des paquets fragmentés (ex : une requête HTTP qui est à cheval sur deux paquets TCP), par des techniques d'évasion d'IDS, etc... La target string est utile, mais pour d'autres buts.

3.15 Logs noyaux : 'Out of window data xxx'

Vous utilisez le patch tcp-window-tracking de patch-o-matic, qui s'occupe de garder trace des paquets acceptables pour les flux TCP autorisés, par rapport au numéro de séquence/accusé de réception, taille de segments, etc... des paquets. Si il détecte qu'un paquet n'est pas acceptable ('out of the window' : en dehors de la fenêtre/fourchette), il le marque invalide et affiche le message de dessus.

Les nouvelles versions enregistrent le paquet et la raison exacte du problème.
- ACK (accusé se réception) est en dessous de la valeur minime (probablement un ACK qui a eu trop de délai),
- ACK (accusé se réception) est au-dessus de la valeur max (accuse la réception de données qu'on a pas encore envoyé),
- SEQ est en dessous du minimum (données déjà accusées de réception sont retransmises),
- SEQ est en dessus du maximum (en dessus de la fenêtre/fourchette de numéros de séquence du receveur).

De plus, dans les nouvelles version, le log peut être totalement supprimé via sysctl :

```
echo 0 > /proc/sys/net/ipv4/netfilter/ip_ct_tcp_log_out_of_window
```

3.16 Pourquoi le code de suivi de connexions suit les connexions marquées UNREPLIED avec un timeout très long ?

Donc vous avez lu et analysé '/proc/net/ip_contrack' et trouvé des entrées UNREPLIED avec un timer très élevé (il peut monter jusqu'à 5 jours), et vous vous demandiez pourquoi gâcher des entrées avec des connexions UNREPLIED (qui ne sont naturellement pas des connexions) ?

La réponse est simple : les entrées UNREPLIED sont des entrées temporaires, c'est-à-dire que dès que l'on n'a plus d'entrées de suivi de connexion de libre, c'est-à-dire que l'on atteint '/proc/sys/net/ipv4/ip_conntrack_max', alors on va effacer quelques vielles entrées UNREPLIED. C'est-à-dire que plutôt que d'avoir des entrées de suivi de connexions vides, on préfère garder des informations qui peuvent de temps en temps être utiles.

3.17 Pourquoi l'option 'iptables -C' (–check) n'est-elle pas implémentée ?

Bon, d'abord, on est fainéant ;-) Pour être honnête, implémenter une option de vérification est pratiquement impossible dés que l'on commence à faire du 'stateful' firewalling. Le traditionnel 'stateless firewalling' base ses décisions juste sur les informations présentes dans l'entête du paquet. Mais le suivi de connexions (et les règles basées sur '-m state'), les résultat du filtrage dépendent de l'en-tête et du contenu du paquet, et aussi de l'en-tête et du contenu des paquets précédents, à l'intérieur de cette connexion.

4 Questions au sujet du développement de Netfilter.

4.1 Je ne comprends pas comment utiliser la target QUEUE à partir du userspace.

Une bibliothèque, appelée 'libipq', est fournie pour gérer les paquets en userspace. Il n'y a pas de documentation pour celle-ci sous forme de pages man. Vous devez compiler et installer les composant de développement de iptables :

```
make install-devel
```

ensuite, aller voir libipq(3).

Vous pouvez être intéressé par les bindings Perl pour libipq, 'Perlipq', qui sont disponibles sur le site

http://www.intercode.com.au/jmorris/perlipq/ . Les bindings eux-même sont un exemple d'utilisation de cette bibliothèque.

D'autre exemples d'utilisation sont :
- testsuite/tools/intercept.c sur le CVS de netfilter,
- ipqmpd (voir http://www.gnumonks.org/projects),
- nfqtest, fait partie de netfilter-tools (voir http://www.gnumonks.org/projects).

4.2 Mon application utilisant libipq me dit "Failed to received netlink message : No buffer space available"

Ça veut dire que la socket Netlink coté noyau n'a plus d'espace mémoire ; le programme en userspace n'est pas capable de gérer la quantité de données délivrées par le noyau.

Est-il possible d'agrandir ces tampons de mémoire noyau pour ne plus avoir ce problème ?

Oui, ce sont des sockets Netlink standards, et vous pouvez régler la taille de leur tampon de réception via /proc/sys/net/core, sysctl ou en utilisant l'option de socket SO_RCVBUF sur le descripteur de fichier.

Vous pouvez aussi faire en sorte que votre application lise les données reçues aussi vite que possible. Si vous n'avez pas besoin du paquet complet, essayez de moins copier vers le userspace (voir ipq_set_mode(3)).

4.3 J'aimerais contribuer un peu de code, mais je n'ai aucune idée de ce qu'il faut faire

L'équipe centrale de Netfilter conserve un fichier 'TODO' ('à-faire') où elle liste tout ce qu'il reste à faire/changer. Vous pouvez retrouver cette liste par anonymous CVS, et les instructions sont sur le site de netfilter.

4.4 J'ai réparé un bug, ou j'ai écrit une extension, comment puis-je la publier ?

Si vous voulez le publier, envoyez le à la liste de diffusion netfilter-devel. Les instructions pour s'abonner se trouvent à cette adresse :

http://lists.netfilter.org/mailman/listinfo/netfilter-devel .

Le meilleur moyen d'envoyer un patch est de suivre les consignes suivantes :
- Sujet commençant par **[PATCH]**
- Le patch doit être inclus directement dans le corps du message, et pas en MIME.

- Mettez une entrée de Changelog, en dehors du diff dans le corps du mail.
- Faîte votre patch sous la forme 'diff -u ancien nouveau', en dehors du répertoire principal (pour qu'il puisse être appliqué avec un '-p1' quand on est dans le répertoire détarré.

Si vous avez écrit une nouvelle extension, ou ajouté des nouvelles options à une vieille extension, une bonne habitude à prendre est de mettre à jour le netfilter-extension-HOWTO pour inclure ces changements, ou pour inclure une description des nouvelles fonctionnalités offertes. De plus, cela attirera plus d'utilisateurs vers votre extension, et vous aurez plus de commentaires en retour en général.