Les Firewalls

Table des matières

Pourquoi un firewall ?

De nos jours, la plus part des entreprises possèdent de nombreux postes informatiques qui sont en général reliés entre eux par un réseau local. Ce réseau permet d'échanger des données entre les divers collaborateurs internes à l'entreprise et ainsi de travailler en équipe sur des projets communs.

La possibilité de travail collaboratif apportée par un réseau local constitue un premier pas. L'étape suivante concerne le besoin d'ouverture du réseau local vers le monde extérieur, c'est à dire internet. En effet, une entreprise n'est jamais complètement fermée sur elle même. Il est par exemple nécessaire de pouvoir partager des informations avec les clients de l'entreprise.

Ouvrir l'entreprise vers le monde extérieur signifie aussi laisser une porte ouverte a divers acteurs étrangers. Cette porte peut être utilisée pour des actions qui, si elles ne sont pas contrôlées, peuvent nuire à l'entreprise (piratage de données, destruction,...). Les mobiles pour effectuer de tel actions sont nombreux et variés : attaque visant le vol de données, passe-temps, ...

Pour parer à ces attaques, une architecture de réseau sécurisée est nécessaire. L'architecture devant être mise en place doit comporter un composant essentiel qui est le firewall. Cette outil a pour but de sécuriser au maximum le réseau local de l'entreprise, de détecter les tentatives d'intrusion et d'y parer au mieux possible. Cela permet de rendre le réseau ouvert sur Internet beaucoup plus sûr. De plus, il peut également permettre de restreindre l'accès interne vers l'extérieur. En effet, des employés peuvent s'adonner à des activités que l'entreprise ne cautionne pas, comme par exemple le partage de fichiers. En plaçant un firewall limitant ou interdisant l'accès à ces services, l'entreprise peut donc avoir un contrôle sur les activités se déroulant dans son enceinte.

Le firewall propose donc un véritable contrôle sur le trafic réseau de l'entreprise. Il permet d'analyser, de sécuriser et de gérer le trafic réseau, et ainsi d'utiliser le réseau de la façon pour laquelle il a été prévu. Tout ceci sans l'encombrer avec des activités inutiles, et d'empêcher une personne sans autorisation d'accéder à ce réseau de données.

Les différentes catégories de firewall

Depuis leur création, les firewalls ont grandement évolué. Ils sont effectivement la première solution technologique utilisé pour la sécurisation des réseaux. De ce fait, il existe maintenant différentes catégories de firewall. Chacune d'entre-elles disposent d'avantages et d'inconvénients qui lui sont propre. Le choix du type d'un type de firewall plutôt qu'un autre dépendra de l'utilisation que l'on souhaite en faire, mais aussi des différentes contraintes imposées par le réseau devant être protégé.

Firewall sans états (stateless)

Ce sont les firewall les plus anciens mais surtout les plus basiques qui existent. Ils font un contrôle de chaque paquets indépendamment des autres en se basant sur les règles prédéfinies par l'administrateur (généralement appelées ACL, Access Control Lists).

Ces firewalls interviennent sur les couches réseau et transport. Les règles de filtrages s'appliquent alors par rapport à une d'adresses IP sources ou destination, mais aussi par rapport à un port source ou destination.

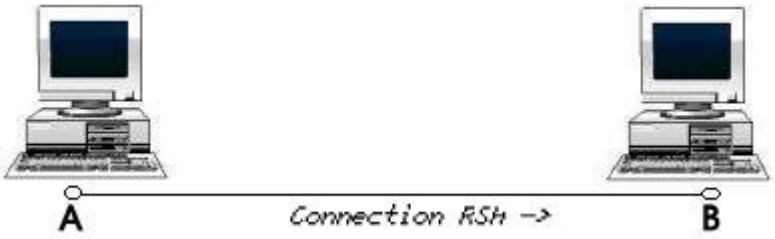
Les limites :

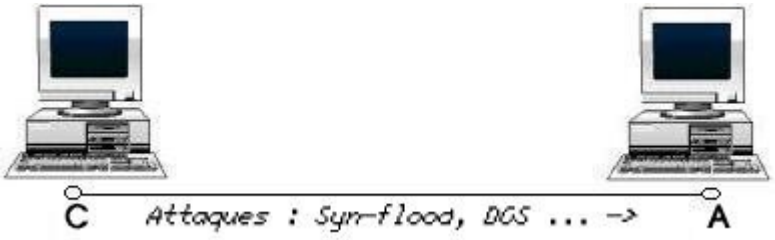
Lors de la création des règles de filtrage, il est d'usage de commencer à spécifier que le firewall ne doit laisser passer aucun paquets. Ensuite, il faut ajouter les règles permettant de choisir les flux que nous souhaitons laisser passer. Il suffit alors d'autoriser l'ouverture des ports des serveurs devant être accessible depuis l'extérieur. Mais les connexions des postes vers l'extérieur poseront problèmes. Effectivement, il faudrait autoriser les ports utilisés par les postes clients lors des connexions vers les serveurs, ceci implique donc d'ouvrir tout les ports supérieurs à 1024. Ceci pose donc un réel problème de sécurité.

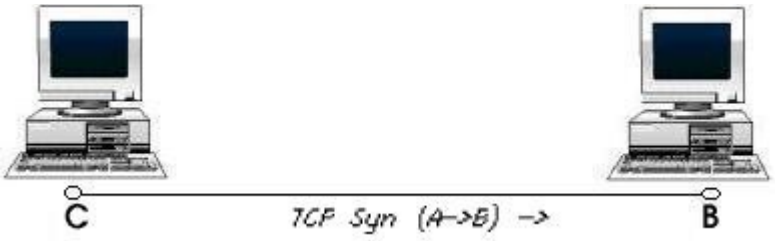
Il n'est pas possible non plus de se préserver des attaques de type ip-spoofing (technique consistant à se faire passer pour une machine de confiance) ou SYN Flood (surcharge de demande de connexion sans attente de la réponse). Les règles de filtrage de ces firewalls sont basées que sur des adresses IP, il suffit donc au pirate de trouver les règles de ce firewall pour pouvoir utiliser cette technique de piratage. Une solution pour se protéger des attaques de type ip-spoofing est de mettre en place une règle interdisant les paquets provenant du réseau extérieur dont l'adresse IP source correspond à une adresse valide du réseau local.

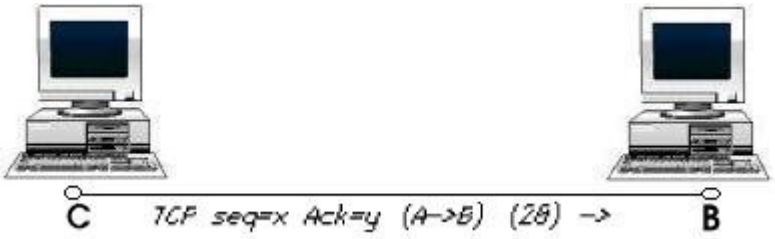
Exemple d'attaque pas ip-spoofing. Une connexion est établie entre le client A et le serveur B. Un

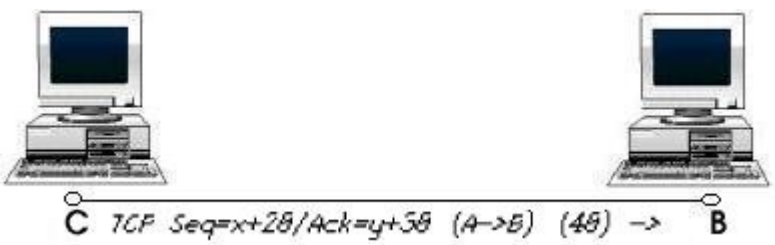
pirate C souhaite attaquer cette connexion.











Une autre limite de ce type de firewall se trouve au niveau des protocoles fonctionnant de manière similaire au FTP. Effectivement, certains protocoles ont besoin d'ouvrir un autre port que celui dédié . Ce port est choisi aléatoirement avec une valeur supérieure à 1024. Dans le cas du protocole FTP, l'utilisation de deux ports permet d'avoir un flux de contrôle et un flux de données pour les connexions. Le problème posé viens du fait que ce port est choisi aléatoirement, il n'est donc pas possible de créer des règles pour permettre les connexions FTP avec les firewalls sans états.

Firewall à états (stateful)

Les firewalls à états sont une évolution des firewalls sans états. La différence entre ces deux types de firewall réside dans la manière dont les paquets sont contrôlés. Les firewalls à états prennent en compte la validité des paquets qui transitent par rapport aux paquets précédemment reçus. Ils gardent alors en mémoire les différents attributs de chaque connexions, de leur commencement jusqu'à leur fin, c'est le mécanisme de stateful inspection. De ce fait, ils seront capables de traiter les paquets non plus uniquement suivant les règles définies par l'administrateur, mais également par rapport à l'état de la session :

- NEW : Un client envoie sa première requête.
- ESTABLISHED : Connexion déjà initiée. Elle suit une connexion NEW.
- RELATED : Peut être une nouvelle connexion, mais elle présente un rapport direct avec une connexion déjà connue.
- INVALID : Correspond à un paquet qui n'est pas valide.

Les attributs gardés en mémoires sont les adresses IP, numéros de port et numéros de séquence des paquets qui ont traversé le firewall. Les firewalls à états sont alors capables de déceler une anomalie protocolaire de TCP. De plus, les connexions actives sont sauvegardées dans une table des états de

connexions. L'application des règles est alors possible sans lire les ACL à chaque fois, car l'ensemble des paquets appartenant à une connexion active seront acceptés.

Un autre avantages de ce type de firewall, se trouve au niveau de la protection contre certaines attaques DoS comme par exemple le Syn Flood. Cette attaque très courante consiste à envoyer en masse des paquets de demande de connexion (SYN) sans en attendre la réponse (c'est ce que l'on appel flood). Ceci provoque la surcharge de la table des connexions des serveurs ce qui les rend incapable d'accepter de nouvelles connexions. Les firewalls stateful étant capables de vérifier l'état des sessions, ils sont capables de détecter les tentatives excessives de demande de connexion. Il est possible, en autre, ne pas accepter plus d'une demande de connexion par seconde pour un client donné.

Un autre atout de ces firewalls est l'acceptation d'établissement de connexions à la demande. C'est à dire qu'il n'est plus nécessaire d'ouvrir l'ensemble des ports supérieurs à 1024. Pour cette fonctionnalité, il existe un comportement différent suivant si le protocole utilisé est de type orienté connexion ou non. Pour les protocoles sans connexion (comme par exemple UDP), les paquets de réponses légitimes aux paquets envoyés sont acceptés pendant un temps donné. Par contre, pour les protocoles fonctionnant de manière similaire à FTP, il faut gérer l'état de deux connexions (donnée et contrôle). Ceci implique donc que le firewall connaisse le fonctionnement du protocole FTP (et des protocoles analogues), afin qu'il laisse passé le flux de données établi par le serveur.

Les limites :

La première limite de ce type de firewall ce situe au niveau du contrôle de la validité des protocoles. Effectivement, les protocoles « maisons » utilisant plusieurs flux de données ne passeront pas, puisque le système de filtrage dynamique n'aura pas connaissance du fonctionnement de ces protocoles particuliers.

Ensuite, il existe un coût supplémentaire lors de la modification des règles du firewall. Il faut que les firewalls réinitialisent leurs tables à état.

Pour finir, ce type de firewall ne protège pas contre l'exploitation des failles applicatives, qui représentent la part la plus importante des risques en terme de sécurité.

Firewall applicatif

Les firewall applicatif (aussi nommé pare-feu de type proxy ou passerelle applicative) fonctionne sur la couche 7 du modèle OSI. Cela suppose que le firewall connaisse l'ensemble des protocoles utilisés par chaque application. Chaque protocole dispose d'un module spécifique à celui-ci. C'est à

dire que, par exemple, le protocole HTTP sera filtré par un processus proxy HTTP.

Ce type de firewall permet alors d'effectuer une analyse beaucoup plus fine des informations qu'ils font transiter. Ils peuvent ainsi rejeter toutes les requêtes non conformes aux spécifications du protocole. Ils sont alors capables de vérifier, par exemple, que seul le protocole HTTP transite à travers le port 80. Il est également possible d'interdire l'utilisation de tunnels TCP permettant de contourner le filtrage par ports. De ce fait, il est possible d'interdire, par exemple, aux utilisateurs d'utiliser certains services, même s'ils changeant le numéro de port d'utilisation du services (comme par exemple les protocoles de peer to peer).

Les limites :

La première limitation de ces firewalls réside sur le fait qu'ils doivent impérativement connaître toutes les règles des protocoles qu'ils doivent filtrer. Effectivement, il faut que le module permettant le filtrage de ces protocoles soit disponible.

Ensuite, ce type de firewall est très gourmand en ressource. Il faut donc s'assurer d'avoir une machine suffisamment puissante pour limiter les possibles ralentissements dans les échanges.

Firewall authentifiant

Les firewall authentifiant permettent de mettre en place des règles de filtrage suivant les utilisateurs et non plus uniquement suivant des machines à travers le filtre IP. Il est alors possible de suivre l'activité réseau par utilisateur.

Pour que le filtrage puisse être possible, il y a une association entre l'utilisateur connecté et l'adresse IP de la machine qu'il utilise. Il existe plusieurs méthode d'association. Par exemple authpf, qui utilise SSH, ou encore NuFW qui effectue l'authentification par connexion.

Firewall personnel

Les firewalls personnels sont installés directement sur les postes de travail. Leur principal but est de contrer les virus informatiques et logiciels espions (spyware).

Leur principal atout est qu'ils permettent de contrôler les accès aux réseaux des applications installés sur la machines. Ils sont capables en effet de repérer et d'empêcher l'ouverture de ports par des applications non autorisées à utiliser le réseau.

Qualité de service

Les firewalls peuvent gérer de la qualité de service pour palier aux limitations du protocole IP. En effet, celui-ci ne fait pas, ou très peu, de différence entre les différents flux. Les données sont traitées de manière équivalente, on utilise le terme « best effort » (effort maximum) pour caractériser IP.

La qualité de service est un terme assez vague qui ne s'applique pas uniquement aux firewalls, et qui varie pour chaque personne et pour chaque usage.

Dans le domaine des réseaux informatiques, la qualité de service est employée pour le contrôle des réseaux en fonction de différents critères qui sont par exemple : la bande passante (débit), les délais (latence) et le taux d'erreurs (perte), …

On pourra donc offrir des qualités de service différentes en fonction de besoins propres à chaque applications (multimédia, transfert de données, …) et en fonction des besoins exprimé par l'utilisateur. Il est donc possible de fournir une qualité de service beaucoup plus fine que la simple limitation matérielle liée au type de la connexion utilisée.

Le firewall pourra donc effectuer de la mise en forme de trafic (« shaping ») pour limiter l'émission de paquets à un débit configuré, réordonnancer (« scheduling ») les paquets afin de prioriser certain flux.

Nous allons maintenant étudier comment configurer Linux à cet effet : les deux principaux éléments intervenant dans cette configuration sont les classificateurs et les files d'attentes. On peut les configurer soit par l'intermédiaire de l'outil nommé « tc » (traffic control) soit par l'intermédiaire du protocole « netlink » pour s'interfacer directement avec le noyau. Nous ne présenterons ici que des exemples basés sur « tc ».

Classificateurs

Dans un gestionnaire de file d'attente basé sur des classes, le classificateur détermine, par l'intermédiaire de filtres (« filter »), dans quelle file d'attente un paquet sera placé.

Les principaux filtres utilisés sont « u32 » et « fw », le filtre « route » est moins utilisé.

Le filtre « fw » s'appuie sur le marquage des paquets par le firewall.

Le filtre « u32 » s'appuie directement sur les données du paquet IP.

Le filtre « route » s'appuie sur la table de routage et permet de prioriser le flux à destination de

certaines routes.

Gestionnaire de file d'attente (« Queueing discipline »)

C'est un algorithme qui gère la file d'attente d'un périphérique réseau. Chaque périphérique est composé d'une file en émission (« egress ») et d'une file en réception (« ingress »).

Ils décident parmi les données, celles qu'il faut envoyer, celles qu'il faut éliminer et celles qu'il faut réordonnancer.

La modification du gestionnaire associé à une file d'attente permet d'en modifier le comportement.

Il existe deux types de gestionnaire de files d'attente : les sans classes (« classless ») et avec classes (« classful »).

Classless

Ce type de gestionnaire est le plus simple, il est dit sans classe car il ne possède pas de subdivisions interne configurable (« class »). Il sera donc impossible de changer le gestionnaire de file d'attentre des subdivisions qui le composent.

Ce type de gestionnaire est généralement utilisé en terminaison des gestionnaires « classful ».

Voici la liste des gestionnaires « classless » existants sous linux :

pfifo_fast

Ce gestionnaire (« First In First Out ») est composé de trois bandes utilisant le champs TOS pour prioriser les paquets. Les paquets seront placé dans l'une des trois bandes en fonction de leur valeur du champ TOS. Les bandes, quand à elles, seront vidées dans l'odre : pour passer à la bande suivante, la bande précédente doit être complétement vide.

La cartographie associant une bande à des valeurs du champs TOS est cependant configurable, voici la cartographie par défaut :

TOS	Bits	Signification	Priorité Linux	Bande
0x0	0	Service Normal	0 Best Effort	1
0x2	1	Minimise le Coût Monétaire (mmc)	1 Filler	2
0x4	2	Maximalise la Fiabilité (mr)	0 Best Effort	1
0x6	3	mmc+mr	0 Best Effort	1
0x8	4	Maximalise le Débit (mt)	2 Masse	2
0xa	5	mmc+mt	2 Masse	2
0xc	6	mr+mt	2 Masse	2
0xe	7	mmc+mr+mt	2 Masse	2
0x10	8	Minimise le Délai (md)	6 Interactive	0
0x12	9	mmc+md	6 Interactive	0
0x14	10	mr+md	6 Interactive	0
0x16	11	mmc+mr+md	6 Interactive	0
0x18	12	mt+md	4 Int. Masse	1
0x1a	13	mmc+mt+md	4 Int. Masse	1
0x1c	14	mr+mt+md	4 Int. Masse	1
0x1e	15	mmc+mr+mt+md	4 Int. Masse	1

Cartographie par défaut

L'illustration ci-dessus met en relation la valeur du champ TOS (colone 1) à la bande associée (colone 5).

tbf

Ce gestionnaire (« Token Bucket Filter », filtre à seau à jetons), très simple, permet de fixer des limites concernant la bande passante. Il permet par ailleurs de définir un débit crête pour dépasser temporairement les limites précédemment fixées.

Il est généralement utilisé pour limiter le trafic sortant et il convient bien pour les bandes passantes importantes.

sfq

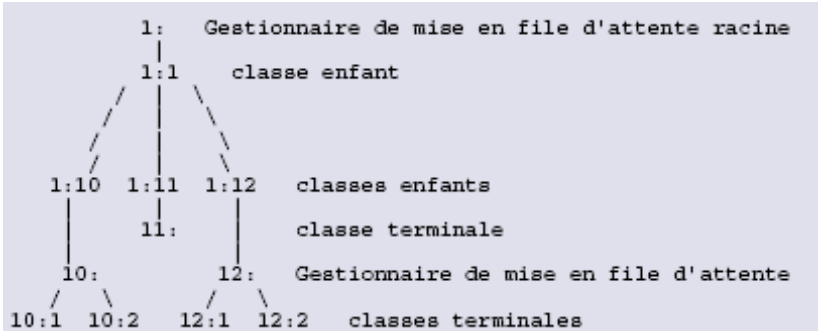
Ce gestionnaire (« Stochastic Fairness Queueing », file d'attente stochastiquement équitable) est très souvent utilisé car il nécessite moins de calculs tout en étant presque parfaitement équitable. En effet, les différents flux de données (sessions TCP par exemple) on la même probabilité d'envoyer des données.

Il est généralement utilisé lorsque le lien est saturé et qu'on souhaite qu'aucune session n'accapare toute la bande passante.

Classful

Ces gestionnaires sont beaucoup plus complexes à configurer car ils font intervenir une notion de parenté avec leurs classes filles qui peuvent à leur tour contenir un gestionnaire de mise en file d'attente.

Le schéma suivant représente un exemple de hiérarchie :



Exemple de hiérarchie classful

prio

Ce gestionnaire est l'équivalent de pfifo_fast précédemment étudié sauf qu'il n'est plus composé de bandes non configurables, mais de classes configurables. Il est donc possible d'avoir plus de trois classes.

cbq

Ce gestionnaire (« Class Based Queueing »), très complexe, peu précis, permet de faire de la mise en forme. Son manque de précision est dû au fait qu'il dépends du taux d'occupation du médium et que le calcul de ce dernier s'avère complexe sur un ordinateur.

htb

Ce gestionnaire (« Hierarchigical Token Bucket », seau de jetons à contrôle hiérarchique) est utilisé pour les mêmes raisons que cbq à la différence près qu'il ne procède pas au calcul du taux d'occupation du médium. Il sera donc le gestionnaire à privilégier. De plus il est plus simple à configurer que son homologue cbq.

Installation sous Linux

Il faut, dans un premier temps, ajouter des options dans le noyau dont le nombre dépend des fonctionnalités qu'on souhaite utiliser.

Il faudra donc cocher les options choisies dans :

```
Networking options  --->
   QoS and/or fair queueing  --->
```

Et dans :

```
Networking options  --->
   QoS and/or fair queueing  --->
   IP: NetFilter Configuration  --->
```

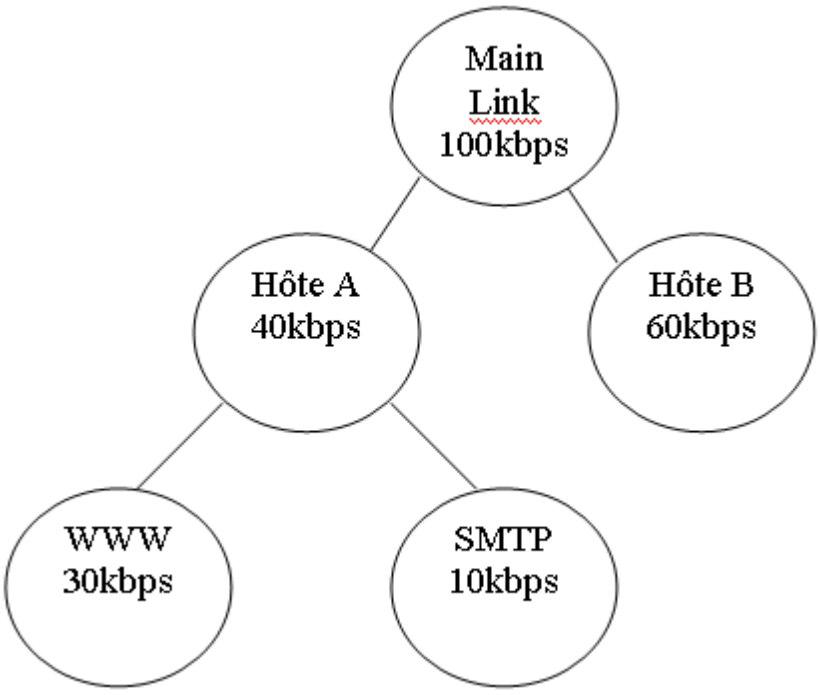
Il faut également installer le paquetage "iproute2" pour avoir accès au logiciel « tc ».

Exemple

Comme nous l'avons indiquer dans la section « classificateur » présenté précédemment, nous pouvons utiliser le firewall pour marquer les paquets et ainsi utiliser ce marquage pour utiliser différents gestionnaires de file d'attente.

Nous souhaitons créer un cannal principal (« main link ») limité à un débit de 100kbps, y allouer à l'hôte A (192.168.0.1) une bande passante de 40kbps et y allouber à l'hôte B (192.168.0.2) une bande passante de 60kbps. L'hôte A pourra utiliser son flux HTTP (www) à un débit de 30kbps et son flux mail (smtp) à un débit de 10kbps.

Il en résulte la représentation schématique suivante :



Représentation schématique

Passons maintenant à la configuration de cet exemple sous linux grace aux outils « tc » et « iptable » :

```
# Création des classes htb
#
# Création de la classe associée au main link, identifiée par '1:1'
$>tc class add dev eth0 parent 1: classid 1:1 htb rate 100kbps ceil 100kbps
# Création de la classe associée à l'hôte A, identifiée par '1:2'
$>tc class add dev eth0 parent 1:1 classid 1:2 htb rate 40kbps ceil 100kbps
# Création de la classe associée au flux www de A, identifiée par '1:10'
$>tc class add dev eth0 parent 1:2 classid 1:10 htb rate 30kbps ceil 100kbps
# Création de la classe associée au flux smpt de A, identifiée par '1:11'
$>tc class add dev eth0 parent 1:2 classid 1:11 htb rate 10kbps ceil 100kbps
# Création de la classe associée à l'hôte B, identifiée par '1:12'
$>tc class add dev eth0 parent 1:1 classid 1:12 htb rate 60kbps ceil 100kbps


# Création des filtres de type fw
#
# Création du filtre utilisant le marquage '1' pour rediriger vers '1:1'
$>tc filter add dev eth0 parent 1: protocol ip prio 1 handle 1 fw classid 1:1
# Création du filtre utilisant le marquage '2' pour rediriger vers '1:2'
$>tc filter add dev eth0 parent 1: protocol ip prio 1 handle 2 fw classid 1:2
# Création du filtre utilisant le marquage '10' pour rediriger vers '1:10'
$>tc filter add dev eth0 parent 1: protocol ip prio 1 handle 10 fw classid 1:10
# Création du filtre utilisant le marquage '11' pour rediriger vers '1:11'
$>tc filter add dev eth0 parent 1: protocol ip prio 2 handle 11 fw classid 1:11


# Positionnement des marquages avec iptable
#
# Positionnement du marquage '1' pour ce qui vient de A
$>iptables -A PREROUTING -i eth0 -t mangle -s 192.168.0.1 -j MARK --set-mark 1
# Positionnement du marquage '11' pour le smtp de A
$>iptables -A PREROUTING -i eth0 -t mangle -s 192.168.0.1 -p tcp --dport 25 -j
MARK --set-mark 11
# Positionnement du marquage '10' pour le www de A
$>iptables -A PREROUTING -i eth0 -t mangle -s 192.168.0.1 -p tcp --dport 80 -j
MARK --set-mark 10
# Positionnement du marquage '2' pour ce qui vient de B
$>iptables -A PREROUTING -i eth0 -t mangle -s 192.168.0.2 -j MARK –ser-mark 2
```

Fonctionnement du pare-feu sous Linux : NetFilter/Iptables

NetFilter est actuellement le filtrage le plus utilisé sous Linux. Il est disponible depuis la version 2.4 du noyau et remplace donc ipchains présent dans la version 2.2. NetFilter est composé de 2 parties : d'une part, NetFilter proprement dit qui doit être compilé dans le noyau (« en dur » ou sous forme de module), d'autre part la commande iptables.

Fonctionnement

Pour opérer le filtrage de paquets, NetFilter stocke un ensemble de règles définies par l'utilisateur. Ces règles sont enregistrées dans des tables sous formes de chaînes. Lorsque NetFilter doit traiter un paquet il applique l'ensemble des règles d'une chaîne les une à la suite des autres. Si le paquet correspond aux critères définis par la règle alors l'action associé à la règle (cible) est effectuée. Dans les paragraphes suivant les principaux types de tables, de chaînes et cibles seront détaillées.

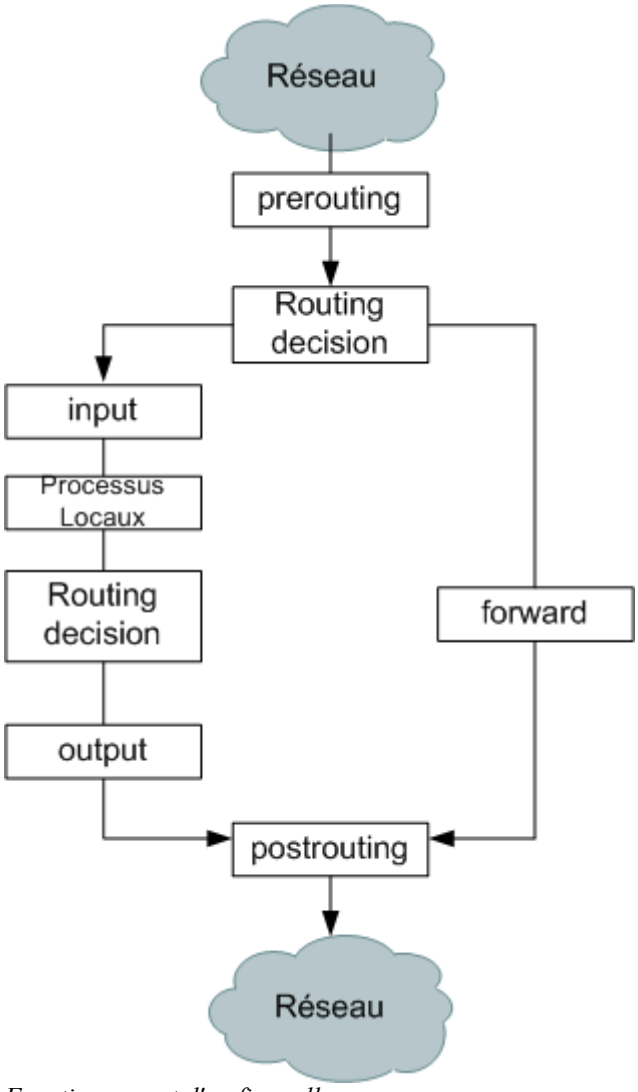
Les tables

Il existe par défaut dans NetFilter une seule table, la table filter. Cette table permet de filtrer les paquets entrant, sortant et transitant avec respectivement les chaînes INPUT, OUPUT et FORWARD. Ces trois chaînes seront détaillées dans le chapitre suivant.

Grâce à l'ajout du module iptable_nat, une nouvelle table est accessible, la table nat. Comme son nom l'indique, elle contient les chaînes qui vont s'appliquer pour la translation d'adresses mais aussi de port. Les chaînes disponible avec ce module sont : PREROUTING, POSTROUTING, OUTPUT. On dispose également de nouvelles cibles notamment MASQUERADE, DNAT, SNAT. Ces deux dernières cible sont respectivement utilisées pour modifier l'adresse destination et l'adresse source des paquets.

Une troisième table disponible est la table MANGLE. Cette table est utilisé notamment lors de la mise en place de la QoS pour marquer les paquets.

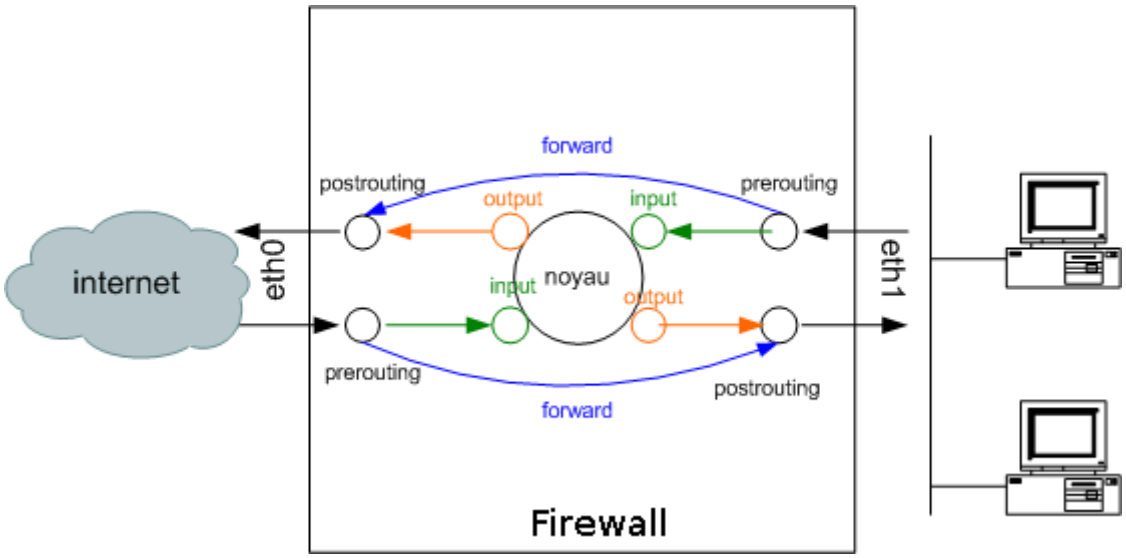
Les chaînes



Fonctionnement d'un firewall

- INPUT : Cette chaîne est utilisée pour les paquets étant à destination des applications du firewall. A ce stade, les paquets sont prêt à être envoyé aux applications.
- OUTPUT : Cette chaîne est utilisée pour les paquets sortant des applications du firewall. A ce stade, les paquets ont donc déjà été traités, ou générés par les applications.
- FORWARD : Cette chaîne filtre les paquets passant d'une interface à une autre du firewall, c'est à dire qu'ils ne sont pas destinés à une application présente sur le firewall. Ces paquets ne passent pas par les chaînes INPUT et OUTPUT et ne passent jamais par la couche applicative. Dans ce cas, le firewall se comportera comme une passerelle.
- PREROUTING : Quand les paquets arrivent au niveau du firewall, ils sont dans un état non modifié. C'est à dire qu'il n'y a encore eu aucun traitement quel qu'il soit sur celui-ci au niveau du firewall. Cette chaîne est utilisée afin de faire des traitements particuliers sur les paquets en arrivé avant d'effectuer leur filtrage à proprement dit. Il est utilisé, par exemple, dans les cas d'utilisation de destination NAT ou DNAT, qui correspond à la modification de l'adresse IP destination.
- POSTROUTING : Quand les paquets sont prêts à être envoyés sur l'interface réseau. Ils ont donc été traités par les applications, et router par le firewall. Tout les traitements sur ces paquets sont alors terminés. Il est utilisé, par exemple, dans le cas de source NAT ou SNAT, qui correspond à la modification de l'adresse IP source (utile pour accéder au réseau Internet avec une adresse IP privé).

Voici comment nous pouvons résumer l'utilisation des chaînes dans un firewall. Nous constatons bien que les chaînes INPUT et OUTPUT sont à destination ou départ du noyau Linux du firewall. Cela valide le fait qu'elle ne sont utilisées que pour les services que firewall lui-même. Nous constatons de même que la chaîne FORWARD ne passe jamais par le noyau du firewall, ces paquets ne sont donc pas traités par les processus externe au firewall.



Les cibles

Un firewall est donc une suite de règles qui spécifient des critères. Si un paquet ne correspond pas à une règle c'est la prochaine règle de la chaîne qui est utilisée, si il correspond la règle va « sauter » (jump) vers une autre règle (target, cible).

Cette cible peut être une autre règle définie par la personne en charge de la configuration du firewall, mais, le plus souvent, ce sont des cibles particulières, définies par Iptables qui sont utilisée.

Parmi les cibles définies par Iptables trois sont fréquemment utilisées : ACCEPT, DROP et REJECT. Ces règles sont dites terminales car elles ne pourront pas être utilisées pour effectuer un saut vers une autre règle.

- ACCEPT signifie qu'on laisse passer le paquet à travers le firewall.
- DROP signifie que le paquet est purement et simplement jeté. L'hôte source du paquet ne sera pas prévenu, le cas est identique à la perte du paquet.
- REJECT signifie que le paquet est rejeté. A la différence de DROP, un paquet d'erreur est transmis à l'émetteur du paquet rejeté. Ainsi celui-ci est prévenu que le paquet a été rejeté et pour donc agir en conséquence.

Exemple de script

Le script ci-dessous est un exemple de script permettant la configuration de NetFilter à l'aide de la commande iptables. Ce type de strict doit être placé à l'emplacement adéquat pour qu'il soit appelé des que les interfaces réseaux sont activées. Par exemple sous Debian il doit se trouver dans le dossier /etc/network/if-pre-up./.

L'exemple fournis ici peut être appliqué à une machine personnelle connecté à Internet via un modem de type PPPoE.

```
#!/bin/sh

#ppp0: internet

#Suppression des règles prédéfinies pour toutes les tables :
iptables -F
iptables -t nat -F
iptables -t mangle -F

#Suppression de toutes les règles de l'utilisateur :
iptables -X

#Politique par défaut (tout rejeter) :
iptables -P INPUT DROP
iptables -P OUTPUT DROP
iptables -P FORWARD DROP

# l'interface loopback du firewall peut émettre dans tous les sens :
iptables -A INPUT -i lo -j ACCEPT
iptables -A OUTPUT -o lo -j ACCEPT

# les connections invalides sont refusées :
iptables -A INPUT -m state --state INVALID -j DROP
iptables -A FORWARD -m state --state INVALID -j DROP

# les connections établies ou assimilables sont acceptées en entrées :
iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT

# Le firewall peut émettre comme il veut sur ppp0 :
iptables -A OUTPUT -o ppp0 -j ACCEPT
```

Les différents types de firewalls

Matériel

Cisco

La célèbre société Cisco, plus connu pour ses routeurs, propose également des firewall matériels nommés PIX (Private Internet eXchange).

Pix 501

Ces firewalls sont des plateformes complètes basées sur un noyau propriétaire de Cisco. Ce sont des firewalls à état (stateful) utilisant l'algorithme de Cisco, l'ASA (Adaptive Security Algorithm). Ils disposent, entre autre, d'un client/serveur DHCP, d'une gestion du PAT (Translation d'Adresse par Port) et NAT (Translation d'Adresse IP), de la prise en compte des réseaux privés virtuel (VPN) avec la gestion d'IPSec. Plusieurs classes de PIX existe, du plus simple pour les petites entreprise, au plus évolué pour les entreprises tel que les fournisseurs d'accès à internet.

De nous jours, ces firewalls tendent de plus en plus à ressembler aux routeurs Cisco. Effectivement, les clients de la marque Cisco réclament fréquemment la mise en place des systèmes d'algorithmes de routage sur les PIX. Ce phénomène existe aussi dans l'autre sens, c'est à dire qu'il existe sur les routeurs Cisco disposant de l'IOS 12 (Internetwork Operating System), le protocole FFS (Firewall Feature Set) de filtrage à état (Stateful).

Pix 515

Voici quelques informations sur le schéma de protection basé sur l'algorithme de sécurité adaptatif (ASA) :

- Adresses IP source et destination
- Numéros de ports source et destination
- Numéros de séquences TCP
- Flags TCP/IP
- Tout paquet sans connexion justifiée est jeté

- Le trafic venant d'une interface de niveau de sécurité haut (inside) vers une interface de niveau bas (outside) est permis, sauf si des access-list (ACL) limitent ce trafic
- Le trafic entrant est interdit par défaut
- Les flux ICMP sont interdits à moins de les permettent spécifiquement.

Logiciels

IPCop

IPCop est un projet Open Source dont le but est d'obtenir une distribution Linux complètement dédiée à la sécurité et aux services essentiels d'un réseau. IPCop joue le rôle d'intermédiaire entre un réseau non sûr (Internet) et un réseau qu'on souhaite sécuriser (réseau local), tout en offrant des services ajoutés.

Les principaux services offerts de base sont les suivants : DHCP, NTP (serveur de temps), PROXY, SSH, IDS(détection d'intrusions), FIREWALL incluant le SHAPING (mise en forme du trafic).

Les services de bases concernant le firewall et le shaping sont assez sommaire et n'exploite pas l'intégralité des fonctionnalités offertes par NetFilter.

IPCop permet l'ajout de fonctionnalités par l'intermédiaire de plugins sans avoir à redémarrer la machine. On peut par exemple citer les plugins suivants : filtrage de mail contre les virus et les spams, filtrage du protocole HTTP et FTP pour les virus, ...

L'installation est simple et rapide car tous les éléments non nécessaires à son objectif de distribution de sécurité ont été omis. Le fait d'omettre un maximum d'éléments est également un gage de sécurité car cela évite au firewall d'être vulnérable aux attaques ciblant des logiciels périphériques (applications bureautique, ...)

La configuration se révèle aussi très simple car elle est effectuée par l'intermédiaire d'une interface web épurée.

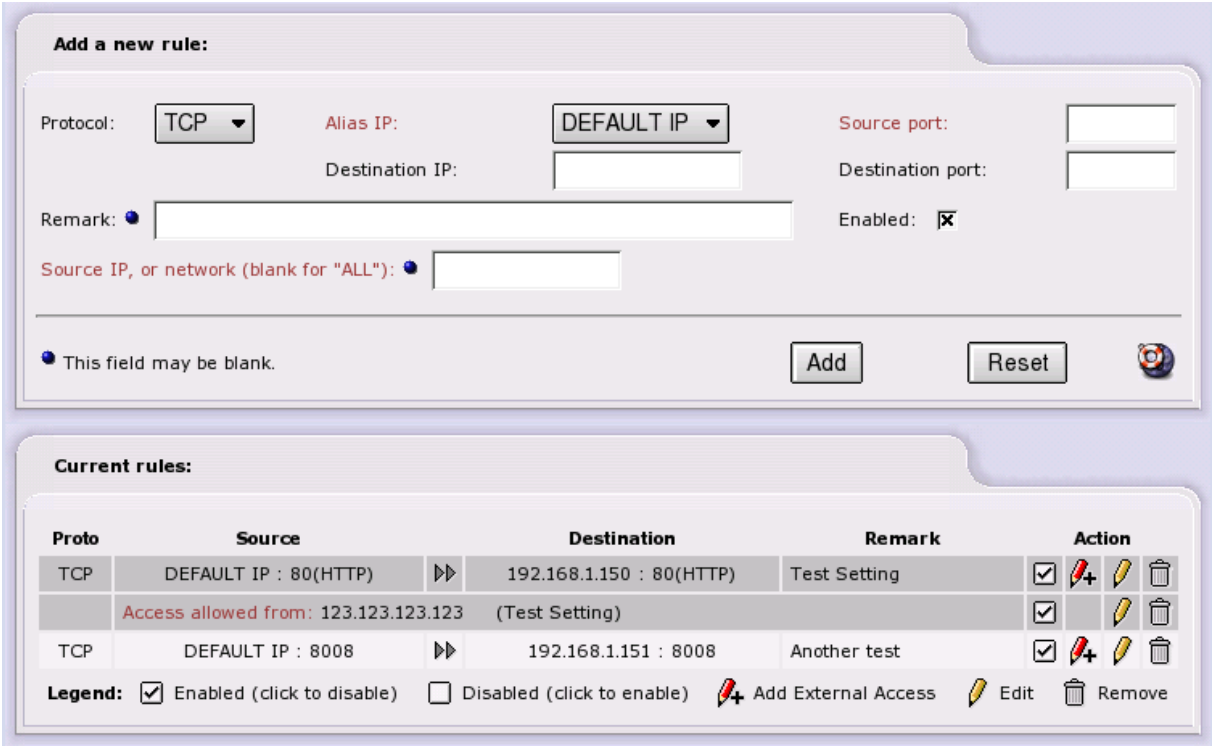
Pour simplifier la configuration, IPCop utilise les bonnes pratiques en terme d'architecture réseau en séparant les réseaux en différentes zones telles que :

- la DMZ : réseau des machines publiant des services services sur Internet. Si une de ces machines est compromise, elle ne pourra pas accéder directement aux machines du LAN.
- le WIRELESS : réseau des machines sans fils, elles ne pourront pas communiquer directement avec les machines du LAN car les réseaux sans fils ne sont pas aussi sécurisées que des liaisons

filaires.

- le LAN: réseau à protéger.
- L'INTERNET: réseau à risque.

L'écran suivant présente une partie de l'interface de configuration d'IPCop :



Ajout d'une nouvelle règle avec IPCop

Conclusion

Comme on peux le constater, les firewalls possèdent de multiple capacités qui peuvent différer en fonction de leurs types. Cette multitude de solutions impose donc une étude rigoureuse de la sécurité devant être mise en place. En effet, le système informatique d'une centrale nucléaire n'aura pas les mêmes besoin en terme de sécurité qu'un particulier et aura donc par conséquent des équipement différents.

Il est également nécessaire de préciser que le firewall est seulement un composant de sécurité, il ne protégera donc pas à lui seul un réseau. Il est nécessaire de l'inclure dans une démarche qui prendra en compte d'autres paramètres tel que la mise à jour des applications.

Sources

Adresse	*Description*
http://www.cisco.com	Site officiel du constructeur Cisco.
http://www.orbytes.fr	Site généraliste sur les réseaux.
http://www.linux-france.org	Site traitant du monde de Linux et du réseau
http://olivieraj.free.fr/fr/linux/information/firewall/	Firewall et sécurité d'un réseau personnel sous Linux
http://fr.wikipedia.org/wiki/Firewall	Firewall, Wikipedia
http://www.frameip.com/firewall/	Les Firewalls par Alban Jacquemin et Adrien Mercier
http://www.netfilter.org/	NetFilter / Iptables
http://www.linux-france.org/prj/inetdoc/	Routage avancé et contrôl de trafic avancé
http://luxik.cdi.cz/~devik/qos/htb/	Gestionnaire HTB
man tc	Page de manuel de la commande tc
man iptables	Page de manuel de la commande iptables