Pare-feu

Chaque ordinateur connecté à Internet (et d'une manière plus générale à n'importe quel réseau) est susceptible d'être victime d'une intrusion pouvant compromettre l'intégrité du système ou bien y altérer les données.
Les pirates informatiques ayant l'intention de s'introduire dans les systèmes informatiques recherchent dans un premier temps des failles, c'est-à-dire une vulnérabilité nuisible à la sécurité du système, dans les protocoles, les systèmes d'exploitations, les applications, voire même le personnel d'une organisation ! Ils scrutent donc le réseau (en envoyant des paquets de données de manière aléatoire) à la recherche d'une machine connectée (machine cible), puis cherchent une faille de sécurité afin de l'exploiter et d'accéder aux données s'y trouvant.
Cette menace est d'autant plus grande que la machine est connectée en permanence à Internet. En effet dans le cas d'une connexion permanente à haut débit :

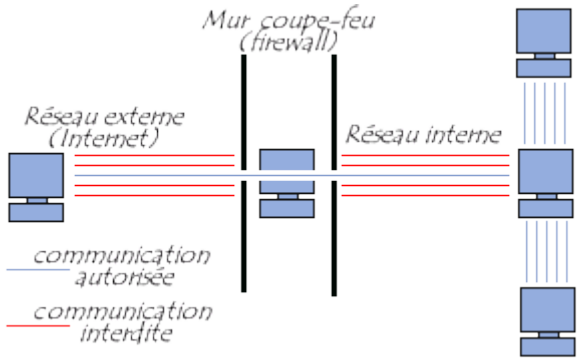
- La machine cible est susceptible d'être connectée sans pour autant être surveillée
- La machine cible est généralement connectée avec une plus large bande passante
- La machine cible ne change pas (ou peu) d'adresse IP

Ainsi, il est nécessaire, notamment pour les entreprises connectées à internet et les internautes ayant une connexion de type câble ou ADSL, de se protéger des intrusions en installant un système pare-feu.

1. Qu'est-ce qu'un pare-feu?

Un pare-feu (firewall en anglais), est un système physique (matériel) ou logique (logiciel) servant d'interface entre un ou plusieurs réseaux afin de contrôler et éventuellement bloquer la circulation des paquets de données, en analysant les informations contenues dans les couches 3, 4 et 7 du modèle OSI. Il s'agit donc d'une machine (machine spécifique dans le cas d'un firewall matériel ou d'un ordinateur sécurisé hébergeant une application particulière de pare-feu) comportant au minimum deux interfaces réseau :

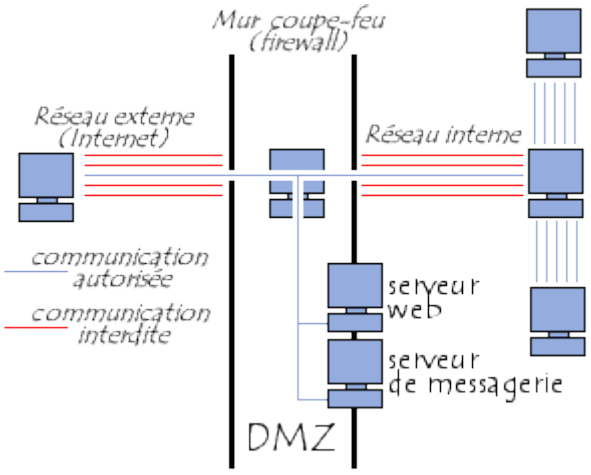
- une interface pour le réseau à protéger (réseau interne)
- une interface pour le réseau externe



Le pare-feu représente ainsi généralement dans les entreprises un dispositif à l'entrée du réseau qui permet de protéger le réseau interne d'éventuelles intrusions en provenance des réseaux externes (souvent internet).

2. Zone Démilitarisée (DMZ)

Lorsque certaines machines du réseau interne ont besoin d'être accessible de l'extérieur (comme c'est le cas par exemple pour un serveur web, un serveur de messagerie, un serveur FTP public, ...) il est souvent nécessaire de créer une nouvelle interface vers un réseau à part, accessible aussi bien du réseau interne que de l'extérieur, sans pour autant risquer de compromettre la sécurité de l'entreprise. On parle ainsi de **zone démilitarisé** (souvent notée **DMZ** pour *DeMilitarized Zone*) pour désigner cette zone isolée hébergeant des applications mises à disposition du public.



3. Principes du filtrage de paquet

Le rôle d'un firewall est d'être un filtre entre le réseau local et un autre réseau. Il se met en place sous la forme d'un routeur ou d'un ordinateur dédié qui transmet les paquets en suivant un certain nombre de règles déterminées.
Il sert aussi à **superviser le trafic** entrant et sortant du réseau, et fournit donc des informations pour détecter des tentatives d'intrusion, ou pour remonter jusqu'aux responsables d'intrusions.
Il ne s'agit pas d'une solution totalement efficace vis à vis des malveillances qu'un réseau local peut subir, mais c'est une façon de se préserver contre certaines tentatives d'intrusion. Elle est par exemple inefficace contre les attaques de virus.

3.1 Le filtrage de paquets simple (Stateless)

Le fonctionnement des systèmes pare-feu, historiquement assuré par les routeurs, est basé sur le principe du filtrage de paquets IP, c'est-à-dire sur l'analyse des en-têtes des paquets IP (aussi appelés *datagrammes*) échangés entre deux machines. En effet les machines d'un réseau relié à Internet sont repérées par une adresse appelée adresse IP.

Ainsi, lorsqu'une machine de l'extérieur se connecte à une machine du réseau local, et vice-versa, les paquets de données passant par le firewall contiennent les en-têtes suivants, qui sont analysés par le firewall:

- L'adresse IP de la machine émettrice
- L'adresse IP de la machine réceptrice
- Le type de paquet (TCP, UDP, ...)
- Le numéro de port (rappel: un port est un numéro associé à un service ou une application réseau)
- le port IP du service demandé
- le port IP du poste demandeur
- le *flag* (drapeau) qui précise si le paquet est une réponse à une demande de service, ou un demande d'établissement de connexion. Un flag ayant la valeur “ACK” (acknowledge) indique que le paquet fait partie d'une connexion en cours, un flag “SYN” définit une ouverture de connexion.

Les adresses IP contenues dans les paquets permettent d'identifier la machine émettrice et la machine cible, tandis que le type de paquet et le numéro de port donnent une indication sur le type de service utilisé. Certains ports sont associés à des service courants (les ports 25 et 110 sont généralement associés au courrier électronique, et le port 80 au Web) et ne sont généralement pas bloqués. Toutefois, il est recommandé de bloquer tous les ports qui ne sont pas indispensables (selon la politique de sécurité retenue).

Le port 23 par exemple est critique car il correspond au service Telnet qui permet d'émuler un accès par terminal à une machine du réseau de manière à pouvoir exécuter des commandes saisies au clavier à distance...

Le **filtre** peut alors mettre en application plusieurs **règles**, contenues dans un fichier de règles, et basées sur les informations des paquets.
Il existe deux façons d'écrire les règles des firewall qui définissent la politique par défaut du filtre:

1. Tout ce qui n'est pas explicitement interdit est autorisé : Dans ce cas, les règles décrivent l'ensemble des services qui doivent être filtrés par le firewall. Les critères de chaque paquet sont comparés aux règles, les unes après les autres, et si aucune ne rejette le paquet, il est accepté.
2. Tout ce qui n'est pas explicitement autorisé est interdit : Dans ce cas, les règles décrivent l'ensemble des services qui doivent être acceptés par le firewall. Les critères de chaque paquet sont comparés aux règles, les unes après les autres, et si l'une accepte le paquet, il est accepté. Cette deuxième méthode est sans nul doute la plus sûre, mais elle impose toutefois une définition précise et contraignante des besoins en terme de communication.

Dans un fichier de règle, il est important de faire attention à l'ordre des règles. Elles sont appliquées les unes après les autres, comme si on appliquait des tamis de plus en plus fins. Donc, si un service est rejeté par une règle, puis accepté par une autre règle plus loin dans le fichier, il ne sera pas accepté.

Exemple de règles sur une interface externe :

No de règle	Interface d'arrivée	Sens	Action	Adresse Source	Port source	Adresse Dest.	Port dest.	Protocole	Description
1	213.152.47.9	Sortie	accepte	195.115.100.2/32	*	*	53	*	accepte les connexions DNS sortantes venant du serveur DNS
2	213.152.47.9	Entrée	accepte	*	53	195.115.100.2/32	*	*	accepte le retour DNS
3	213.152.47.9	Entrée	accepte	*	*	195.115.100.3/32	80	TCP	accepte le trafic http entrant à destination du serveur de la DMZ
4	213.152.47.9	Sortie	accepte	195.115.100.3/32	80	*	*	TCP	accepte le trafic http sortant en provenance du serveur de la DMZ
5	213.152.47.9	Sortie	accepte	192.168.0.0/24	>1024	*	80	TCP	accepte le trafic http sortant à destination du d'internet venant du réseau local
6	213.152.47.9	Entrée	accepte	*	80	192.168.0.0/24	>1024	TCP	accepte le trafic http entrant à destination du serveur de la DMZ
défaut	213.152.47.9		bloque	toutes	tous	toutes	tous	tous	Règle par défaut, tout ce qui n’est pas autorisé est interdit

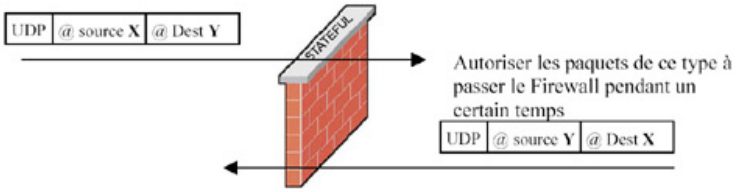
Nous remarquons dans ce tableau de règles que pour autoriser les postes du réseau local à accéder au Web, on est obliger de laisser ouverts tous les ports supérieur à 1024. On pourrait améliorer cette règle en spécifiant que le champs flag doit être différent de “SYN” pour bloquer les ouvertures de connexion et ne laisser passer que les paquets correspondant à une connexion établie.

3.2 Le filtrage dynamique et adaptatif (Stateful)

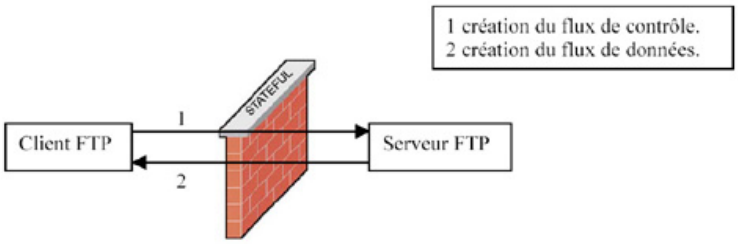
Le fonctionnement décrit ci-dessus ne s'attache qu'à examiner statiquement les champs du datagramme IP. Or, de nombreux services (le FTP par exemple) initient une connexion sur un port statique, mais ouvrent dynamiquement (c'est-à-dire de manière aléatoire) un port afin d'établir une session entre la machine faisant office de serveur et la machine cliente. Ainsi, il est impossible de prévoir les ports à laisser passer ou à interdire. Pour y remédier, l'entreprise *Check point* a breveté un système de **filtrage dynamique de paquets** (le terme anglo-saxon exact étant *stateful inspection*) basé sur l'inspection des couches 3 et 4 du modèle OSI, permettant d'effectuer un suivi des transactions entre le client et le serveur et donc d'assurer la bonne circulation des données de la session en cours.

L'amélioration par rapport au filtrage simple, est la conservation de la trace des sessions et des connexions dans des tables d'états internes au Firewall. Le Firewall prend alors ses décisions en fonction des états de connexions, et peut réagir dans le cas de situations protocolaires anormales. Ce filtrage permet aussi de se protéger face à certains types d'attaques DoS (Denial Of Service).

Dans l'exemple précédent, on va autoriser l'établissement des connexions à la demande, ce qui signifie que l'on aura plus besoin de garder tous les ports supérieurs à 1024 ouverts. Pour les protocoles Udp et Icmp, il n'y a pas de mode connecté. La solution consiste à autoriser pendant un certain délai les réponses légitimes aux paquets envoyés. Les paquets Icmp sont normalement bloqués par le Firewall, qui doit en garder les traces. Cependant, il n'est pas nécessaire de bloquer les paquets Icmp de type 3 (destination inaccessible) et 4 (ralentissement de la source) qui ne sont pas utilisables par un attaquant. On peut donc choisir de les laisser passer, suite à l'échec d'une connexion Tcp ou après l'envoi d'un paquet Udp.



Pour le protocole Ftp (et les protocoles fonctionnant de la même façon), c'est plus délicat puisqu'il va falloir gérer l'état de deux connexions. En effet, le protocole Ftp, gère un canal de contrôle établi par le client, et un canal de données établi par le serveur. Le Firewall devra donc laisser passer le flux de données établi par le serveur, qui est associé au flux de contrôle. Ce qui implique que le Firewall connaisse le protocole Ftp, et tous les protocoles fonctionnant sur le même principe.



Exemple de règles sur une interface externe avec un parefeu dynamique:

No de règle	Interface d'arrivée	Sens	Action	Adresse Source	Port source	Adresse Dest.	Port dest.	Protocole	etat connexion TCP	Description
1	213.152.47.9	Entrée	accepte	*	*	195.115.100.3/32	80	TCP	*	accepte le trafic http entrant à destination du serveur de la DMZ
2	213.152.47.9	Sortie	accepte	192.168.0.0/24	>1024	*	80	TCP	*	accepte le trafic http sortant à destination du d'internet venant du réseau local
3	213.152.47.9	Sortie	accepte	192.168.0.0/24	>1024	*	21	TCP	*	accepte le trafic ftp sortant à destination du d'internet venant du réseau local
4	213.152.47.9	Sortie	accepte	*	*	*	*	TCP,UDP	Etablie, associée	accepte tout flux sortant correspondant à une connexion établie ou associée à une autre connexion (ftp)
5	213.152.47.9	Entrée	accepte	*	*	*	*	TCP	Etablie, associée	accepte toute flux entrant lié à une connexion établie ou associée à une autre connexion (ftp).
6	213.152.47.9	Sortie	accepte	*	_	*	_	ICMP	*	accepte le trafic ICMP sortant
7	213.152.47.9	Entrée	accepte	*	_	*	_	ICMP	Etablie, associée	accepte les paquets ICMP entrants associé à un flux TCP ou UDP
défaut	213.152.47.9		bloque	*	*	*	*	*	*	Règle par défaut, tout ce qui n'est pas autorisé est interdit

Si le filtrage dynamique est plus performant que le filtrage de paquets basique, il ne protège pas pour autant de failles applicatives, c'est-à-dire les failles liées aux logiciels, représentant la part la plus importante des risques en terme de sécurité.

3.3 Le filtrage applicatif (ou pare-feu de type proxy)

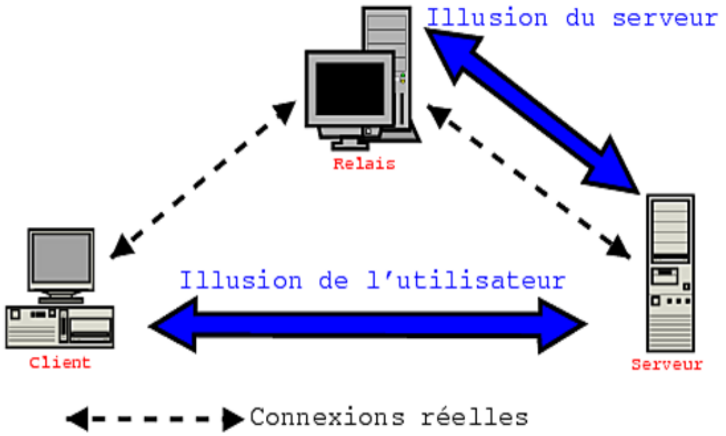
Le filtrage applicatif permet, comme son nom l'indique, de filtrer les communications application par application, ce qui signifie qu'il travaille au niveau de la couche 7 du modèle OSI. Le filtrage applicatif suppose donc une connaissance de l'application, et notamment de la manière de laquelle elle structure les données échangées.

Un firewall effectuant un filtrage applicatif est appelé *passerelle applicative* ou serveur mandataire (proxy en anglais) car il permet de relayer des informations entre deux réseaux en effectuant un filtrage fin au niveau du contenu des paquets échangés. Il s'agit donc d'un dispositif performant assurant une bonne protection du réseau, pour peu qu'il soit correctement administré. En contrepartie une analyse fine des données applicatives requiert une grande puissance de calcul et se traduit donc souvent par un ralentissement des communications.

L'exemple le plus connu est le Proxy HTTP. Mais il existe d'autres types de serveurs proxy pour d'autres protocoles (SMTP, POP3, IMAP, FTP)

Son rôle :

- Il **relaye** tous les échanges entre clients et serveurs
- Il **masque** certaines informations (nom et adresse IP) des machines du réseau interne, et les remplace par les siennes
- Il maintient un **journal** de tout ce qui passe par lui.
- Il **filtre** les informations échangée en fonction de **critères** : adresses, ports, filtrage de contenu (scripts, applets java, ActiveX,...)n filtrage sémantique (mot-clés, nom de domaines, url)
- Il peut inclure une fonction **cache** : par exemple pour un proxy HTTP, les pages web récemment consultées sont mises en **cache**, ce qui accélère les consultations. On parle quelquefois de proxy-cache. Plusieurs proxy caches peuvent coopérer pour synchroniser leurs caches.
- Il peut s'interfacer avec un **antivirus**



Le proxy permet donc d'empêcher les personnes extérieures au réseau d'avoir des informations sur l'organisation du réseau, comme par exemple les adresses IP des postes. Seule celle du serveur Proxy sera connue de l'extérieur.
Par contre, les applications doivent être prévues pour utiliser un Proxy, et l'utilisateur doit configurer ses applications localement avec les informations du Proxy. Rendre son utilisation transparente nécessite un redirection de port sur le parefeu

4. la traduction d'adresse

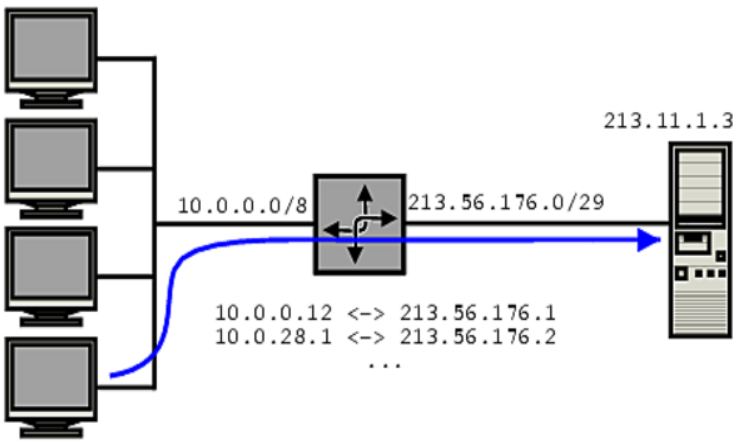
La traduction d'adresse permet de substituer d'autres valeurs aux adresses et ports source ou destination dans les paquets traversant un routeur.
Ce mécanisme a deux application principales : la redirection qui permet de rediriger un flux entrant vers une autre adresse et éventuellement sur un autre port, et le camouflage IP (ip masquerading) qui permet de substituer à un adresse IP source du réseau local, une adresse IP publique afin de permettre aux utilisateurs d'un réseau IP privé d'accéder au réseau publique Internet, sans que leur adresse soit vue de l'extérieur.
Dans les deux cas, la traduction d'adresse opérée dans un sens sera inversée dans l'autre sens.

4.1 Redirection d'adresse

La redirection d'adresse et de port permet par exemple de rediriger les flux entrant sur l'interface externe sur un port donné vers un serveur situé à l'intérieur du réseau local. Par exemple, le flux reçu à destination de l'adresse publique du routeur et du port 80 sera redirigé vers l'adresse IP du serveur Web interne au réseau local.

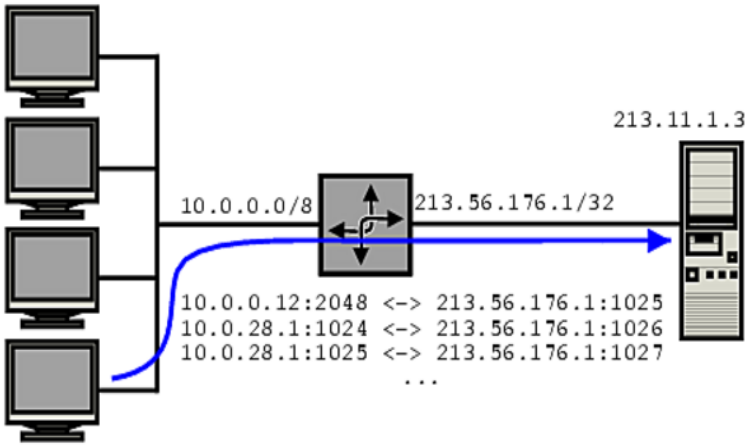
4.2 Network Address Translation - NAT

Le routeur établit et maintient une correspondance entre les adresses internes (généralement privées pour éviter un effet de masquage) et une ou plusieurs adresses publiques. Cette correspondance peut être statique ou bien réalisée dynamiquement par l'équipement. Dans ce cas, le nombre d'hôtes pouvant sortir simultanément est limité par le nombre d'adresses publiques utilisables



4.3 Port Address Translation - PAT

Pour contourner la limitation qui précède, la correspondance n'est plus établi sur la simple adresse IP mais utilise un couple adresse IP/port (TCP/UDP) ou adresse IP/ID (ICMP). C'est une technique couramment utilisée pour connecter un réseau à l'aide d'une unique machine ayant un compte chez un FAI .



5. Autres fonctions

L'évolution des pare-feu a conduit à l'ajout de fonctionnalités dont le domaine peut sembler connexe. Parmi celles-ci ont peut distinguer :

- les réseaux privés virtuels « VPN » : les possibilités proposées vont de la création d'un « extranet » (réseau interne multi-site utilisant des tunnels chiffrants entre sites) à la sécurisation de l'accès aux ressources internes des itinérants ;
- la haute disponibilité, synchronisation des tables de filtrage
- enfin certains équipements se proposent d'inclure des filtres du niveau applicatif, comme un antivirus, la recherche de contenus licencieux, une sonde de détection d'intrusion,...Cela se fait généralement au prix d'une consommation de ressources (recherches de signatures) qui peut grever les performances globales, et cela contrevient au principe de minimisation de la taille du code pour minimiser les risques de faille résiduelle.

6. Les limites des firewalls

Le fait d'installer un firewall n'est bien évidemment pas signe de sécurité absolue.

Les firewalls ne protègent en effet que des communications passant à travers eux. Ainsi, les accès au réseau extérieur non réalisés au travers du firewall sont autant de failles de sécurité. C'est par exemple le cas des connexions effectuées à l'aide d'un modem. D'autre part, le fait d'introduire des supports de stockage provenant de l'extérieur sur des machines internes au réseau peut être fort préjudiciable pour la sécurité de ce dernier.

La mise en place d'un firewall doit donc se faire en accord avec une véritable politique de sécurité. D'autre part la mise en place d'un système pare-feu n'exempt pas de se tenir au courant des failles de sécurité et d'essayer de les minimiser...