Les Pare-feux (Firewalls)

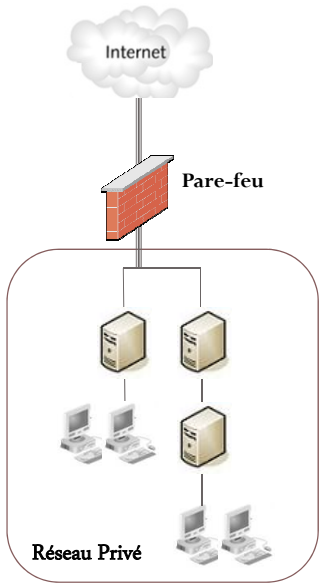
Dr. Khaled Hamouid
Département d'informatique
Université de Batna 2
k.hamouid@univ-batna2.dz

Introduction

- Un domaine à protéger
  - Un réseau interne (*sûre*) ne fait pas confiance à un réseau externe (*non sûre*)
  - Ex: Un réseau d'entreprise/personnel que l'on veut protéger
    - Prévenir des attaques provenant d'un réseau externe
    - Protéger l'accès aux services internes
- Comment ?
  - Appliquer une politique de contrôle d'accès entre les deux réseaux
    - ***Systèmes Pare-feux (Firewalls)***

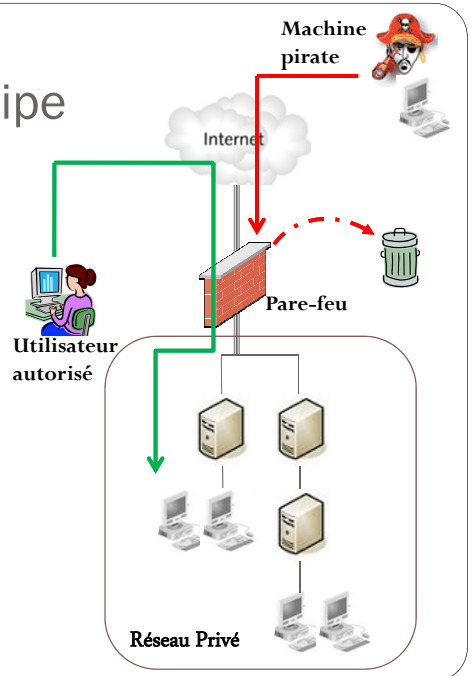
Pare-feux : Principe

- Définition
  - Ensemble de composants installés entre deux réseaux et appliquant une politique d'accès
    - En général, entre un réseau à protéger (interne) et un réseau non sécuritaire (externe)



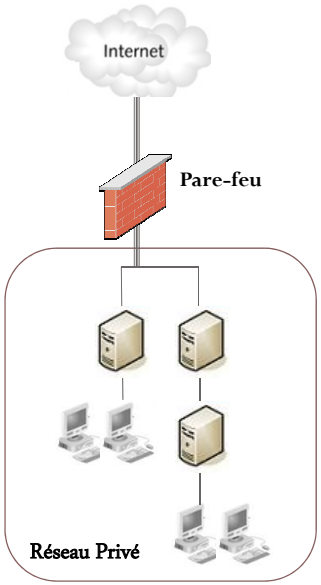
Pare-feux : Principe

- Propriétés
  - Tout le trafic transitant entre les deux réseaux passe nécessairement par le pare-feu ;
  - Seul le trafic explicitement autorisé par les règles du pare-feu appliquée est autorisé à passer au travers ;



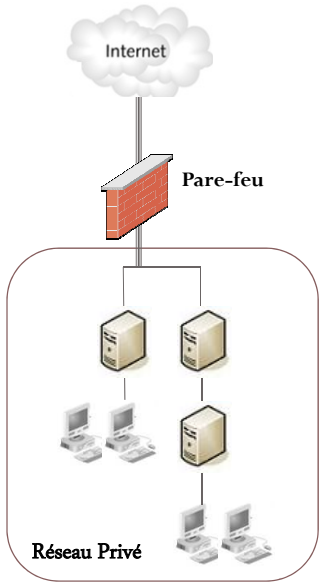
Pare-feux : Principe

- Fonctions
  - Contrôler l'accès entrant et sortant
    - Inspecter les communications entre les deux réseaux
    - Agir sur le trafic transitant selon une politique de sécurité prédéfinie
    - Une politique de sécurité est interprétée par un ensemble de règles de filtrage
  - Auditer/tracer de façon "centrale" le trafic, aider à prévoir les évolutions du réseau (statistiques possibles)



Pare-feux : Principe

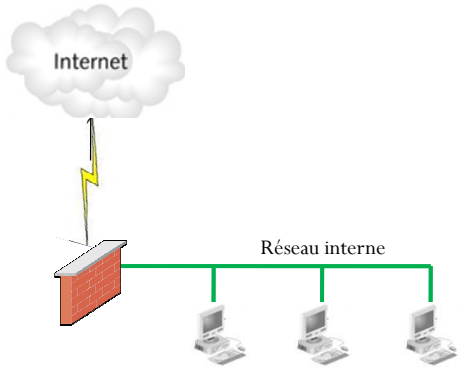
- Un pare-feu ne peut pas prévenir:
  - Attaques qui ne passent pas par lui
  - Intrusions internes
  - Les virus



Différentes topologies de déploiement des Pare-feux

- Architecture avec un seul pare-feu central
- Architecture à plusieurs pare-feu
- Architecture avec zone démilitarisée (DMZ)

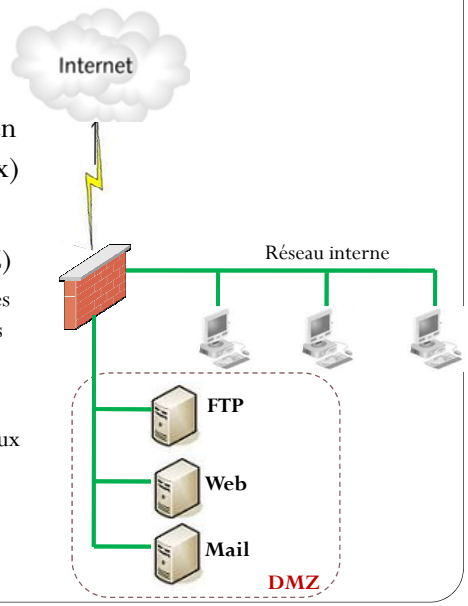
Architecture avec un seul pare-feu central



Architecture avec zone démilitarisé (DMZ)

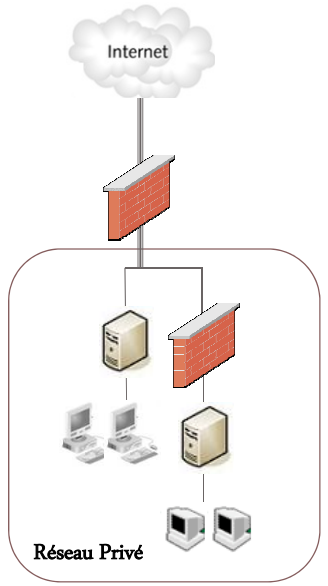
Cette approche consiste à diviser le réseau interne en deux parties (sous réseaux)

- Réseau des serveurs (DMZ)
  - un semble de services destinés aux utilisateurs externes mais protégés par le pare-feu.
- Réseau interne
  - le réseau des utilisateurs locaux est séparé de la DMZ.



Architecture à double pare-feux

- Ajouter un second pare-feux entre deux segments du réseau interne (segmentation à double pare-feux)
- Protéger un segment sensible (finance, production…)
- Sécurité renforcée: l'accès entre le segment protégé et les autres segments du réseau interne est contrôlé par le deuxième pare-feu
  - Protection contre des attaques internes



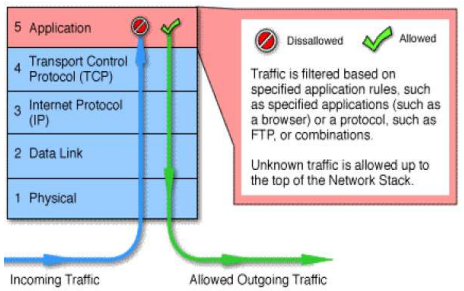
Types de pare-feu

Principalement deux types de pare-feux :

- Pare-feu filtre de paquets : niveau Réseau du modèle OSI
  - *filtrage à état (statefull)*
  - *filtrage sans état (stateless)*
- Pare-feu applicatif (Proxy) : niveau application du modèle OSI

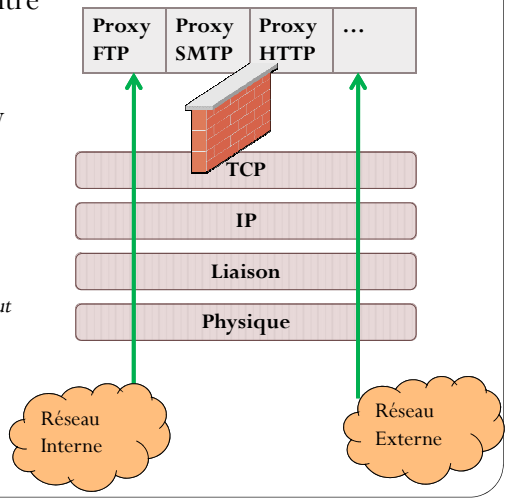
Pare-feu applicatif (Proxy)

- Analyse du trafic échangé au niveau application (niveau 7 du modèle OSI)
  - Appliquer une politique de sécurité spécifique de chaque application
  - Un Proxy est nécessaire : Chaque application de la couche application (Telnet, FTP, SMTP, HTTP, etc.) est associée à un Proxy



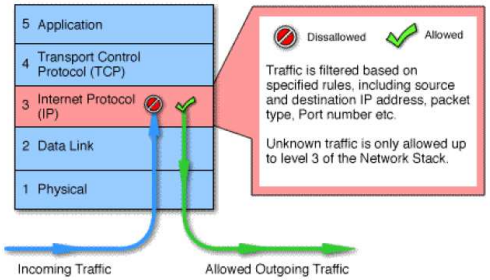
Pare-feu applicatif (Proxy)

- **Proxy** : un élément intermédiaire obligatoire entre le client et le serveur
- Les règles de filtrage sont implémentées dans le proxy
- Décision propre à une application :
  - Un utilisateur ne peut que uploader des fichiers
    - seules les requêtes de type *ftp put* sont autorisées.
  - Seules les requêtes HTTP de lecture de type *http get* sont autorisées.



Pare-feu niveau réseau

- Il analyse le trafic de données au niveau de la couche réseau (Niveau 3 du modèle OSI)
- Appliquer un ensemble de règles à chaque paquet IP entrant et sortant et décide s'il doit être transmis sur le réseau ou détruit.
- Le filtrage est généralement configuré pour filtrer les paquets dans les deux sens (depuis et vers le réseau interne)



Pare-feu niveau réseau

- Les règles de filtrage sont basés sur les informations se trouvant dans l'en-tête des paquets IP :
  - le port source/destination
  - le protocole UDP/TCP
  - l'adresse source/destination
  - le type de paquet ICMP
  - la taille du paquet
  - interface réseau d'entrée / de sortie
  - les drapeaux (flags) de connexion TCP ;

Pare-feu niveau réseau

- **Filtrage de paquets sans état (*Stateless*)**
  - Pas de suivi de connexion
    - Examiner les paquets indépendamment les uns des autres
  - Filtrage basé sur les informations brutes de l'entête des paquets IP
- **Filtrage de paquets à état (*Stateful*)**
  - Suivi de connexion
  - Filtrage basé sur l'état de la connexion
    - Ce paquet est-il une réponse ?
    - Ce paquet est-il relié à une connexion déjà établie ?
    - …
  - Plus puissant que le pare-feu *Stateless*

Politique de sécurité et règles de filtrage

- **Une politique de sécurité :** Ensemble de règles et de procédures élaborées par la direction en collaboration avec des professionnels de la sécurité
  - Acceptables/inacceptables utilisations du réseau
  - Identification des ressources à protéger
  - Comment réagir aux brèches de sécurité
- La politique de sécurité dépend des besoins de sécurité
  - Accès à partir de l'extérieur est limité au serveur Email
  - Les utilisateurs internes sont autorisés à accéder au réseau externe uniquement vers un service HTTP

Politique de sécurité et règles de filtrage

- La politique de sécurité définit l'ensemble des règles de filtrage à implémenter sur le pare-feu
- **Règle de filtrage :** les règles sont une série de critères auquel doivent ou non répondre les paquets. Si un paquet satisfait les critères d'une règle alors cette règle est appliquée.
  - Les différentes règles de filtrage sont appliquées les unes à la suite des autres.
  - A chaque règle est associée une action à effectuer si la règle doit s'appliquer
    - Accept, Drop, log, etc,

Politique de sécurité et règles de filtrage

- **Forme d'une règle de filtrage :**

  SI <*condition*> ALORS <*action*>

<*condition*> : La condition s'exprime en fonction des critères de sélection qui sont dans le cas de filtrage de paquets, les champs de l'entête IP (adresse IP, port, protocole, etc.)

<*action*> : – **Accept** (autoriser le paquet),
– **Drop** (interdire le paquet)

- **Politique par défaut d'un pare-feu :**

Définit l'action par défaut à appliquer lorsque aucune règle n'est appliquée

- Deux stratégies:
  - Tout ce qui n'est pas explicitement autorisé est interdit (interdire par défaut tous les paquets)
  - Tout ce qui n'est pas explicitement interdit est autorisé (autoriser par défaut tous les paquets)

Filtrage de paquets : Exemple de règles

- Traduire la politique de sécurité en des règles précises concernant des communications IP

Politique A) : on fait pas confiance à l'hôte 10.5.5.5, on autorise la réception des Email mais uniquement par le serveur SMTP

Action	IP source	Port Source	IP Destination	Port Destination
Drop	10.5.5.5	*	Mes Hôtes	*
Accept	*	*	Ma passerelle	25

Politique B) : interdire tout par défaut

Action	IP source	Port Source	IP Destination	Port Destination
Drop	*	*	*	*

Filtrage de paquets : Exemple de règles

Politique C) :

- On autorise la réception des Emails sur notre serveur smtp (1.2.3.4)
- On autorise tous les utilisateurs internes à se connecter à tous les services du réseau externe
- Rien d'autres

Action	IP source	Port Source	IP Destination	Port Destination
Accept	*	*	1.2.3.4	25
Accept	Mes hôtes	*	*	*
Drop	*	*	*	*

Problème : les règles de filtrage ne respectent pas la politique sus-définie

Filtrage de paquets : Exemple de règles

Problème de la Politique C) : Échec des connexions sortantes

Les hôtes internes ouvrent une connexion TCP sur le port 80 :

- Le paquet initial Syn est autorisé par la règle 2
- Le paquet Syn\Ack de retour est rejeté par la règle 3

Solution : autoriser la réception des paquets associés à une connexion sortante (les paquets avec le flag ACK activé).

Action	IP source	Port Source	IP Destination	Port Destination	Flag
Accept	*	*	1.2.3.4	25	
Accept	Mes hôtes	*	*	*	
Accept	*	*	Mes hôtes	*	ACK
Drop	*	*	*	*	

Filtrage de paquets : Netfilter/iptables

Netfilter : un module dans le noyau Linux 2.4 destiné au filtrage de paquets.

Iptables : Langage (interface) pour spécifier les règles de filtrage avec Netfilter

Chaque paquet inspecté passe à travers une suite de tables préconstruites, Une table est un ensemble de chaînes de traitement elles-mêmes composées de règles.

Il existe 3 tables (Filter, NAT et Mangle)

La table Filter : c'est la table qui contient toutes les règles de filtrage de paquets, ces règles sont regroupées en trois chaines :

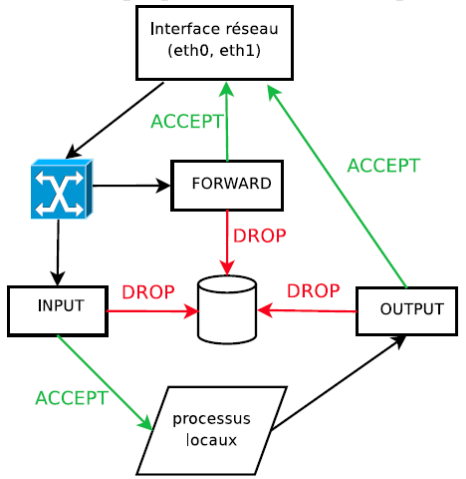
INPUT : pour les paquets entrants

OUTPUT : pour les paquets sortants

FORWORD : pour les paquets qui passe d'une interface réseau à l'autre

Filtrage de paquets : Netfilter/iptables

- Cycle de vie d'un paquet traversant un pare-feu iptables



Filtrage de paquets : Netfilter/iptables

Syntaxe d'une règle du pare-feu iptables :

iptables -A <chaîne> -i <interface d'entrée> -o <interface de sortie> -p <protocole> -s <IP source> -d <IP destination> --sport <port source> --dport <port destination> -j <action>

<chaîne> : chaîne de filtrage (INPUT, OUTPUT, FORWORD)

<interface d'entrée \ de sortie> : interface réseau par laquelle le paquet passe (exp : eth0,eth1,ppp0)

<protocole> : le protocole de la couche 3 ou 4 utilisé dans la communication (exp : tcp, icmp, udp)

<IP source> \ <IP destination> : l'adresse IP source et destination

<port source> \ <port destination> : numéro de port source et destination identifiant l'application associée à la communication

<action> : l'action à appliquer sur le paquet lorsque celui-ci satisfait les critères de la règle (deux actions : DROP, ACCEPT)

Filtrage de paquets : Netfilter/iptables

Exemples de règles de filtrage :

iptables -A INPUT -j DROP : Supprime tous les paquets rentrants

iptables -A OUTPUT -j ACCEPT : Autorise tous les paquets sortants

iptables -A INPUT -p tcp -sport 80 -j DROP : Supprime toutes les trames HTTP rentrant dans l'espace utilisateur.

iptables -A INPUT -p tcp -sport ! 80 -j DROP : Supprime toutes les trames rentrant excepté celles de HTTP.

Définition de la politique par défaut : dans les trois chaines tous ce qui n'est pas explicitement autorisé est strictement interdit

iptables -P INPUT DROP
iptables -P OUTPUT DROP
iptables -P FORWARD DROP

Filtrage de paquets : Netfilter/iptables

Remise à zéro du pare-feu: effacer toutes les règles

iptables -F INPUT DROP
iptables -F OUTPUT DROP
iptables -F FORWARD DROP

Filtrage de paquets : Netfilter/iptables

Suivi de connexion :

Capacité d'un pare-feu de filtrer les paquets en fonction de l'état de la connexion au quelle sont associés (*filtrage statefull*)

Iptables est capable de distinguer entre trois états :

NEW : paquet demandant une nouvelle connexion

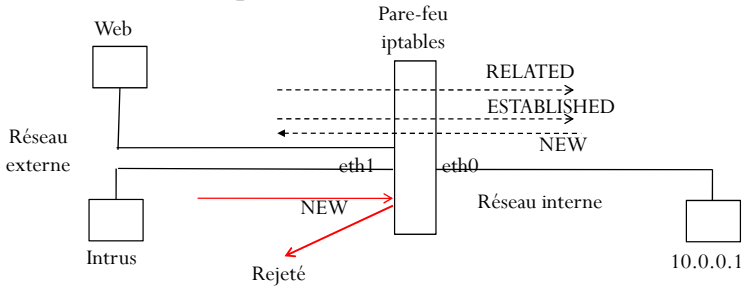
ESTABLISHED : paquet associé à une connexion déjà établie

RELATED : Nouvelle connexion mais liée, idéal pour les connexions FTP

INVALID : paquet associé à une connexion inconnue

Filtrage de paquets : Netfilter/iptables

Suivi de connexion (Exemple) :



Autoriser les connexions sortantes initialisées par la machine interne, rejeter toute connexion rentrante sauf celles en réponse avec les premières :

```
iptables -A FORWARD -o eth1 -s 10.0.0.1 -d 0.0.0.0/0 -p all --
state ! INVALID
-j ACCEPT

iptables -A FORWARD -i eth1 -s 0.0.0.0/0 -d 10.0.0.1 -p all –
state RELATED, ESTABLISHED -j ACCEPT
```

Filtrage de paquets : Netfilter/iptables

Protection de certaines attaques :

IP spoofing : refuser les paquets entrants prétendant provenir de notre propre réseau

```
iptables -A INPUT -i NET -s LAN -j DROP
iptables -A FORWARD -i NET -s LAN -j DROP
```

TCP Syn flooding : réduire la réception des paquets Syn par unité de temps

```
iptables -A INPUT/FORWARD -p tcp --state ! INVALID --limit 1/second -j
ACCEPT
```

Smurf :

```
iptables -A INPUT -p icmp --icmp-type echo-reply --limit 1/second
-J ACCEPT
iptables -A FORWARD -p icmp -d 192.168.0.255 -j DROP
```