Mise en place du pare-feu

Choix du système d'exploitation

Nous avons le choix entre les systèmes Windows ou Linux que nous comparons dans le tableau ci-dessous.

	Windows (7)	**Linux (Debian Jessie)**
Processeur	1 GHz	1 GHz
Mémoire Vive (RAM)	2 Go	512 Mo
Espace disque	20 Go	2 Go

Pour sa légèreté et le contrôle total sur le système, nous avons choisi d'utiliser une distribution Linux. Notre choix s'est porté sur Debian pour sa stabilité et sa communauté très active.

Choix du pare-feu

Une fois notre système d'exploitation installé sur notre machine physique, nous procédons à la sélection du logiciel tenant le rôle de pare-feu. Parmi les sélectionner, nous trouvons : IPCop, IPTables, pfSense et Shorewall.
Nous avons retenu les critères suivant : les **performances** du système, la facilité de **prise en main** et la présence d'une **communauté**. Chaque critère est évalué sur 10 points.

	IPCop	**IPTables**	**pfSense**	**Shorewall**
Performance	7	10	8	7
Prise en main	8	7	9	7
Communauté	7	10	7	9
Moyenne	7	9	8	8

IPTables obtient la moyenne la plus haute de notre comparatif. Il fonctionne directement sur le noyau du système lui permettant d'obtenir des performances imbattable. De plus, il est dépourvu d'options supplémentaire pouvant venir surcharger le programme. Malgré tout, il reste plus complexe à prendre en main, il se gère intégralement en ligne de commande et il est nécessaire de connaître une syntaxe particulière pour accepter les commandes et atteindre l'objectif souhaité.

IPCop et pfSense ne nécessite pas l'installation d'un système d'exploitation étant une distribution compléte. Cependant, il reste basé sur Linux et garde les même pré-requis.

Configuration du pare-feu

IPTables étant pré-installé sur notre distribution, nous pouvons procéder directement à sa configuration. Nous mettons les règles de bases de notre IPTables 

:

```
# iptables -P INPUT DROP
# iptables -P FORWARD DROP
```

```
# iptables -P OUTPUT ACCEPT

# iptables -A INPUT -i lo -j ACCEPT
# iptables -A INPUT -p tcp --dport 22 -j ACCEPT
# iptables -A INPUT -m conntrack --ctstate RELATED,ESTABLISHED -j ACCEPT
# iptables -A INPUT -p icmp --icmp-type 0 -j ACCEPT
# iptables -A INPUT -p icmp --icmp-type 8 -j ACCEPT
# iptables -A INPUT -j DROP
```

Règles de filtrage

```
# iptables -I FORWARD 1 -n conntrack --ctstate RELATED,ESTABLISHED
```

-n conntrack permet de suivre l'état d'une connexion
–ctstate permet de vérifier l'état de la connexion

On active l'*IP Forwarding* de manière permanente et on ajoute le NAT sur l'interface de sortie (eth2) pour masquer les IPs de notre réseau local :

```
# sed -i 's,net.ipv4.ip_forward=0,net.ipv4.ip_forward=1,g' /etc/sysctl.conf
# sysctl -p /etc/sysctl.conf
# iptables -t nat -A POSTROUTING -o eth2 -j MASQUERADE
```

Configuration des routes

En plus des règles par défaut, on ajoute les routes nécessaires au bon fonctionnement de notre réseau :

```
route add -net 172.25.0.0 netmask 255.255.255.0 gw 172.31.2.2
route add -net 192.168.0.0 netmask 255.255.0.0 gw 172.31.2.2
```

Ces deux lignes ont était incluse dans le fichier de configuration de nos interface **/etc/network/interface**.

⇒ Configuration des interfaces
⇒ Configuration des règles IPTables