université de BORDEAUX

TP6 (partie B)

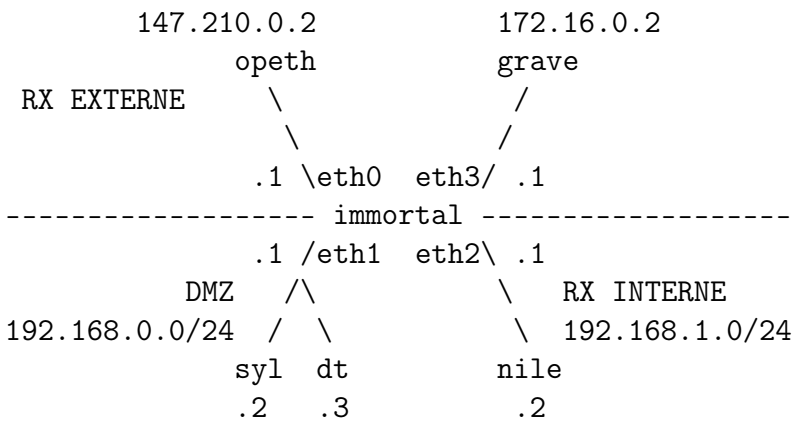
Important : Pour réaliser ce TP à distance au CREMI, il faudra commencer par trouver un poste libre (par exemple, *miro* dans la salle 008), le réveiller s'il est éteint et ensuite s'y connecter avec la commande *ssh* (plus d'info). Par ailleurs, il faudra adapter les commandes *QemuNet* pour utiliser un affichage en mode texte basé sur *tmux*, surtout si votre connexion n'est pas fiable... Voir la documentation de *QemuNet* sur la page Moodle, ainsi que ce petit tutoriel en vidéo.

1 Firewall

Lancez la configuration *QemuNet* suivante[1] :

```
$ /net/ens/qemunet/qemunet.sh -x -s /net/ens/qemunet/demo/dmz.tgz
```

Voici la topologie de notre réseau :



Les IP et les tables de routage sont déjà configurées pour vous :-) Par défaut, il n'y a aucun firewall en place, ce qui signifie que tout le monde peut communiquer avec tout le monde sans restriction particulière.

Dans cet exercice, nous jouons le rôle de l'administrateur d'un petit réseau d'entreprise, rélié à Internet par la passerelle *immortal*. "Internet" est ici représenté par seulement les deux machines `opeth` est `grave`. Nous allons configurer un firewall sur cette passerelle à l'aide de la commande `iptables`. Un memento dans la section suivante vous donne la syntaxe de cette commande pour vous aider, commencez par y jeter un coup d'oeil.

1. Adaptez la commande si vous souhaitez lancer *QemuNet* à distance avec *tmux*.

D'autre part, rapidement vous aurez besoin d'utiliser `tcpdump` pour vérifier à quel endroit quels paquets parviennent à passer ou restent au contraire bloqués, utilisez-le vraiment abondamment !

À noter que `iptables` est en train d'être supplanté par `nftables`, mais dans un futur proche, c'est encore `iptables` que vous verrez en production, et les principes sont les mêmes de toutes façons.

— Au sein d'un réseau d'entreprise, quel différence y-a-t'il entre la DMZ et le réseau interne des employés ?
— Sur *immortal*, positionnez la politique par défaut à DROP :
```
iptables -P INPUT DROP
iptables -P OUTPUT DROP
iptables -P FORWARD DROP
```
— Nous venons donc d'activer le firewall sur *immortal*. Plus aucun trafic réseau n'est autorisé vers ou à travers *immortal*. Vérifiez avec ping.

Nous allons maintenant ajouter des règles pour autoriser explicitement seulement certains trafics réseau. Toutes les questions suivantes nécessitent donc de taper des commandes (sur *immortal* uniquement) de la forme : `iptables -A FORWARD ... -j ACCEPT` , cf le memento plus bas pour les détails de syntaxe. Si vous avez fait une erreur, vous pouvez supprimer une ligne en reprenant la même commande et en remplaçant `-A` par `-D`

— Autoriser le ping (c'est-à-dire le protocole icmp) du réseau Interne vers Internet, sans autoriser l'inverse.
— Faire un test *ping*, constater que cela ne fonctionne pas : il faut effectivement autoriser la réponse à passer ! Pour simplifier l'autorisation des réponses de manière générale, on peut utiliser
`iptables -A FORWARD -m state --state RELATED,ESTABLISHED -j ACCEPT`
qui laissera passer tout ce qui est identifié comme réponse à ce que l'on a déjà laissé passer auparavant (c'est donc un "catch-all" pour les réponses).
— Autorisez l'accès au web depuis les machines du réseau interne. Faire un test avec *wget*.
— Autorisez *grave* à accèder au serveur ssh (port 22) de *dt*. Faire un test avec le compte `toto` (mot de passe `toto`).
— Autorisez l'accès depuis n'importe où vers le serveur web de syl (port 80). Faites un test avec wget.
— Depuis opeth et grave, testez votre firewall sur la DMZ avec nmap !

Memento Firewall

La configuration du firewall se base sur la table "filter" de la commande `iptables`. Elle est subdivisée en 3 chaînes, notée <CHAIN> :

— INPUT : tout ce qui est à destination de la machine elle-même ; cela ne concerne donc pas les paquets qui seront relayés
— OUTPUT : tout ce qui est émis par la machine elle-même ; cela ne concerne donc pas les paquets qui seront relayés
— FORWARD : tout ce qui ne fait que traverser la machine (donc les paquets relayés).

Voici la syntaxe de principales commandes `iptables` :

- Pour afficher les règles de la table filter : `iptables -L` , on peut y ajouter l'option `-v` pour voir les statistiques, pour vérifier par exemple qu'une règle attrape bien des paquets.
- Pour effacer toutes les règles ajoutées : `iptables -F`
- Pour chaque règle que l'on ajoute, trois actions sont possibles (notée <ACTION>) :
  - ACCEPT : on accepte ;
  - REJECT : on rejette poliment (réponse d'erreur envoyé à l'émetteur) ;
  - DROP : on jette à la poubelle (pas de réponse d'erreur).
- Pour modifier la politique par défaut du firewall :
  `iptables -P <CHAIN> <ACTION>`
- Pour ajouter une nouvelle règle à une chaîne du firewall :
  `iptables -A <CHAIN> <SRC> <DST> <...> -j <ACTION>`
  - avec <SRC> des indications éventuelles sur la provenance des paquets IP, comme par exemple `-i eth0` ou `-s 192.168.0.0/24` ;
  - avec <DST> des indications éventuelles sur la destination des paquets IP, comme par exemple : `-o eth1` ou `-d 147.210.0.0/24` ;
  - avec <...> des infos complémentaires par exemple sur la nature du protocole `-p icmp` , `-p tcp` , ou `-p udp` avec après éventuellement des précisions spécifiques à ces protocoles (`--dport 80` pour TCP) ou encore sur l'état `-m state --state NEW`, ...

  - sur l'état, il est notamment utile d'employer `-m state --state RELATED, ESTABLISHED` pour identifier les paquets qui sont une réponse à quelque chose que l'on a déjà laissé passer : une réponse à un ping, une réponse à une demande de connexion TCP, etc.
  - Pour supprimer une règle, on peut utiliser `iptables -D <CHAIN>` en mettant derrière soit un numéro de règle (lisible avec `iptables -L --line-numbers`, soit la règle elle-même (i.e. remplacer `-A` par `-D`).
  - Pour supprimer toutes les règles, on peut utiliser `iptables -F <CHAIN>`

Pour plus d'info, consulter le manuel : `man iptables` et `man iptables-extensions`