Architecture sécurisée - TD

Éric Leblond

28 novembre 2006

Introduction

Le but de ce TD est de travailler sur les capacités de filtrage de GNU/Linux. On abordera ensuite la mise en place d'un IPS.

1 Utilisation de Netfilter

1.1 Introduction

La couche Netfilter intégrée au noyau propose une couche de filtrage complète permettant de réaliser des politiques de filtrages avancées.

1.2 Mise en place de la politique

On mettra tout d'abord en place une politique par défaut stricte :

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

Une fois cette politique établie on vérifiera que tout trafic réseau est impossible. On lancera en particulier un ping vers localhost.

Question 1 *Rétablir le trafic vers localhost. On expliquera chacune des règles employées.*

1.2.1 Ajout des logs

```
# iptables -A INPUT -j LOG --log-prefix 'input drop'
# iptables -A FORWARD -j LOG  --log-prefix 'forward drop'
# iptables -A OUTPUT -j LOG  --log-prefix 'output drop'
```

1.2.2 Ouverture du trafic HTTP sortant

On autorisera le trafic sortant à destination du port 80.

Question 2 *Pourquoi la réception est-elle impossible ? On analysera les échanges grâce à tcpdump et aux logs.*

1.2.3 Positionnement des règles statefull

Question 3 *Mettre en place les régles relatives au trafic établi en INPUT, FORWARD et OUTPUT.*

1.3 Manipulation de régles

Lister les règles avec la commande :

```
# iptables -L -nv
```

Question 4 *Supprimer la règle introduite pour autoriser le trafic à destination du port 80.*

1.4 Ajout de régle

Question 5 *Ajouter une règle permettant de résoudre les noms*

Question 6 *En utilisant le module multiport ajouter une règle permettant de sortir vers les ports 3128,80,21,443*

1.5 Prise en charge de la translation d'adresse

Question 7 *On redirigera le trafic à destination du port 80 à destination la machine vers 192.168.200.100. La connectivité sera vérifiée en utilisant* `tcpdump`

On listera ensuite l'ensemble des connexions en utilisant `/proc/net/ip_conntrack`.

Question 8 *Pourquoi as-t-on deux couples d'adresses IP ?*

1.6 Politique avancée

Question 9 *En utilisant une négation, limiter le trafic de l'utilisateur root au réseau local*

La méthode précédente nous oblige à connaitre le réseau local. Il peut donc être intéressant de supprimer cette dépendance :

Question 10 *En utilisant la target TTL, limiter le trafic de l'utilisateur root au réseau local*

2 Utilisation de snort-inline

`snort-inline` est un IPS basé sur snort. Il permet de récupérer les paquet depuis `libipq` au lieu d'utiliser `libpcap`. Snort-inline est donc capable de bloquer les paquets et agit donc comme un IPS.

On installera `snort-inline` disponible à l'adresse suivante :

2.1 Installation

Les sources de `snort-inline` peuvent être récupérées à l'adresse suivante :

```
http://snort-inline.sourceforge.net/
```

Après extraction de l'archive on installera le logiciel dans `/usr/local` avec des fichiers de configuration dans `/etc/snort-inline`.

```
./configure --enable-nfnetlink --prefix=/usr/local --sysconfdir=/etc/snort-inline/
```

Parmi les dépendances de compilation on a :
- libpcre3-dev
- libnfnetlink
- libnetfilter_queue
- libdnet : http ://libdnet.sourceforge.net/

On copiera les fichiers du répertoire `etc/` dans le répertoire `/etc/snort-inline`.

2.2 Configuration

On mettra en place `snort-inline` de manière à valider le fonctionnement au moyen de règles écrites pour l'occasion.

Question 11 *Écrire les règles de filtrage permettant de vérifier tous les paquets à destination et en provenance de TCP port 80.*

Des signatures sont nécessaires au bon fonctionnement de `snort-inline`.

On va générer un fichier `custom.rules` que l'on référencera dans le fichier `snort_inline.conf`

```
alert tcp any any -> $HOME_NET 80 (msg:"TCP port 80";)
alert tcp $HOME_NET any -> any 80 (msg:"TCP client port 80";)
alert tcp any any -> any 80 (msg:"global TCP port 80";)
```

On ajustera $HOME_NET de manière ce qu'il soit égal à l'IP de la machine de tests.

2.3 Tests

Un fois la configuration effectuée, on peut lancer `snort_inline` :

```
sudo snort_inline -H 1  -Q   -l /tmp/snort -K ascii \
    -c /etc/snort-inline/snort_inline.conf
```

On validera le fonctionnement en essayant d'emettre et de recevoir des paquets à destination du port 80. On observera notamment les journaux dans `/tmp/snort`.

Après avoir ajouter la règle suivante, on validera qu'il n'est plus possible de récupérer des contenus.

```
drop tcp any any -> $HOME_NET 80 (msg:"TCP port 80"; content:"GET";)
```