SADEK
ADEL
SIO1

Tp 4 iptables

Mémo : input et output dst ou src c'est que le routeur forward c'est pour le routage

J'allume ma machine dhcpServeur,RouteurDhcp,client1Dhcp(qui héberge un serveur DNS)

Mon routeur à une troisième patte dans le réseau de la classe

Adresse ip de mon routeur dans le réseau de la classe : 172.16.18 .236 (eth2)

```
source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto eth0
iface eth0 inet static
        address 172.20.128.254
        netmask 255.255.255.0
auto eth1
iface eth1 inet static
        address 192.168.128.254
        netmask 255.255.255.0

auto eth2
iface eth2 inet static
        address 172.16.18.236
        netmask 255.255.255.0
        gateway 172.16.18.254

up /root/configRoute_Gab_Cora.sh
up iptables -t nat -A POSTROUTING -o eth2 -j MASQUERADE

root@debTP3:~# *
```

Voici le fichier de conf de mon routeur

Je vérifie si mon srvDhcp arrive à pinger le routeur qui est dans la salle

Le ping depuis mon client 1 dans mon réseau interne vers le routeur de la salle marche parfaitement

Donc le routage fonctionne correctement (après avoir rajouter la règle de NAT)

Iptables -t nat -A POSTROUTING -o eth2 -j MASQUERADE

```
root@debTP3:~# ping 172.16.18.254
PING 172.16.18.254 (172.16.18.254) 56(84) bytes of data.
64 bytes from 172.16.18.254: icmp_seq=1 ttl=254 time=0.740 ms
64 bytes from 172.16.18.254: icmp_seq=2 ttl=254 time=1.22 ms
64 bytes from 172.16.18.254: icmp_seq=3 ttl=254 time=0.992 ms
^C
--- 172.16.18.254 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 13ms
rtt min/avg/max/mdev = 0.740/0.984/1.220/0.196 ms
```

```
#!/bin/sh

iptables -F INPUT
iptables -F OUTPUT
iptables -F FORWARD
iptables -t nat -F POSTROUTING
iptables -t nat -F PREROUTING

echo "Souhaitez vous affichez le contenue de toutes les tables ?"
read reponse

if [ $reponse == oui ]
then

iptables -L
iptables -t nat -L

fi
```

Je crée le script parefeu_off.sh pour vider toutes les tables quand je le souhaite

La commande pour vider une table est iptables -F INPUT (ou OUTPUT,FORWARD)

J'ai fait le script demander j'ai rajouté aussi une option qui me permet d'afficher le contenue de toute les tables pour être sur que tout a été supprimer

Je crée le script parefeu_on.sh

Je configure la politique de sécurité de sorte a ce qu'elle soit en mode DROP sur toutes mes chain input,output,forward, tout paquet sera interdit de passer appart ceux qui seront explicitement autorisé

```
root@debTP3:~# iptables -L
Chain INPUT (policy DROP)
target     prot opt source               destination

Chain FORWARD (policy DROP)
target     prot opt source               destination

Chain OUTPUT (policy DROP)
target     prot opt source               destination
root@debTP3:~#
```

Pour modifier la politique de sécurité la commande est iptables -P INPUT DROP

Je teste maintenant le scripts parefeu_off.sh

Je me rends compte que la politique ne revient pas en mode ACCEPT je modifie mon script pour qu'elle repasse en ACCEPT

Mon script parefeu_off ressemble a sa maintenant

```
#!/bin/sh

iptables -F INPUT
iptables -F OUTPUT
iptables -F FORWARD
iptables -t nat -F POSTROUTING
iptables -t nat -F PREROUTING
iptables -P INPUT ACCEPT
iptables -P FORWARD ACCEPT
iptables -P OUPUT ACCEPT
echo "Souhaitez vous affichez le contenue de toutes les tables ?"
read reponse

if [ $reponse = oui ]
then

iptables -L
iptables -t nat -L
```

Je re-test

Sa fonctionne parfaitement

```
root@debTP3:~# ./parefeu_off.sh
iptables: Bad built-in chain name.
Souhaitez vous affichez le contenue de toutes les tables ?
oui
Chain INPUT (policy ACCEPT)
target     prot opt source               destination

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy DROP)
target     prot opt source               destination
Chain PREROUTING (policy ACCEPT)
target     prot opt source               destination

Chain INPUT (policy ACCEPT)
target     prot opt source               destination

Chain POSTROUTING (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
```

Je vais autoriser toutes les communications pour l'interface « lo » local

J'ai rajouté cette commande

```
iptables -A INPUT -i lo -j ACCEPT
iptables -A OUTPUT -o lo -j ACCEPT
```

Je teste un ping vers ma loopback avant pour voir si sa ping pas

Et ça ne fonctionne pas

```
root@debTP3:~# ping 172.16.18.236
PING 172.16.18.236 (172.16.18.236) 56(84) bytes of data.
ping: sendmsg: Opération non permise
ping: sendmsg: Opération non permise
ping: sendmsg: Opération non permise
ping: sendmsg: Opération non permise
ping: sendmsg: Opération non permise
```

Je lance le script
Je vérifie avec iptables -L si elle a bien été prise en compte
La règle a bien été ajouter

```
root@debTP3:~# iptables -L
Chain INPUT (policy DROP)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere

Chain FORWARD (policy DROP)
target     prot opt source               destination

Chain OUTPUT (policy DROP)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere
root@debTP3:~#
```

Je test le ping

```
root@debTP3:~# ./parefeu_on.sh
root@debTP3:~# iptables -L
Chain INPUT (policy DROP)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere

Chain FORWARD (policy DROP)
target     prot opt source               destination

Chain OUTPUT (policy DROP)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere
root@debTP3:~# ping 172.16.18.236
PING 172.16.18.236 (172.16.18.236) 56(84) bytes of data.
64 bytes from 172.16.18.236: icmp_seq=1 ttl=64 time=0.027 ms
64 bytes from 172.16.18.236: icmp_seq=2 ttl=64 time=0.146 ms
64 bytes from 172.16.18.236: icmp_seq=3 ttl=64 time=0.049 ms
^C
--- 172.16.18.236 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 56ms
rtt min/avg/max/mdev = 0.027/0.074/0.146/0.051 ms
root@debTP3:~#
```

Ça fonctionne

J'arrête le pare feu avec mon script pour arrêter

Je vais autoriser l'accès en ssh sur ma machine sur toute les interface et pour le flux aller et retour

```
#!/bin/sh

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

iptables -A INPUT -i lo -j ACCEPT
iptables -A OUTPUT -o lo -j ACCEPT

iptables -A INPUT -i eth0 -p tcp --dport 22 -j ACCEPT
iptables -A OUTPUT -o eth0 -p tcp --sport 22 -j ACCEPT


iptables -A INPUT -i eth1 -p tcp --dport 22 -j ACCEPT
iptables -A OUTPUT -o eth1 -p tcp --sport 22 -j ACCEPT

iptables -A INPUT -i eth2 -p tcp --dport 22 -j ACCEPT
iptables -A OUTPUT -o eth2 -p tcp --sport 22 -j ACCEPT
```

Je vais essayer de me connecter depuis ma machine client 1 en ssh sur mon routeur

Quand nous sommes en iptables et que l'on veut dire « tous » il ne faut rien dire ne rien préciser si on ne précise pas d'interface d'entrée par default il appliquera la règle sur toute les interfaces d'entrée

La connexion fonctionne parfaitement

```
root@debTP3:~# ssh root@172.20.128.254
The authenticity of host '172.20.128.254 (172.20.128.254)' can't be established.
ECDSA key fingerprint is SHA256:5Shvi8y7JDXfcojUkBksNtbFxmi8LtCHcn4AgzuHgXg.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '172.20.128.254' (ECDSA) to the list of known hosts.
root@172.20.128.254's password:
Linux debTP3 4.19.0-12-amd64 #1 SMP Debian 4.19.152-1 (2020-10-18) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sun Dec 13 22:40:52 2020
```

Je vais interdire l'accès depuis un client, j'ai bien placer l'interdiction avant l'autorisation général pour ne pas que la règle générale englobe la plus affinée

```
iptables -A INPUT -p tcp --dport 22 -s 172.20.128.25 -j DROP

iptables -A INPUT  -p tcp --dport 22 -j ACCEPT
iptables -A OUTPUT -p tcp --sport 22 -j ACCEPT
```

Comme ceci je vide les chain et je lance mon script parefeu_on et je revérifie si je peux me connecter en ssh

Ça ne fonctionne pas :

```
root@debTP3:~# ssh root@172.20.128.254
ssh: connect to host 172.20.128.254 port 22: Connection timed out
root@debTP3:~#
```

Je vais autoriser seulement l'accès au serveur DHCP pour cela je commente toute les règle en lien avec le ssh et je bloque tout accès en ssh puis j'autorise le serveur DHCP a ce connecter en ssh

Sa marche parfaitement

```
root@srvdhcp:~# ssh root@172.20.128.254
The authenticity of host '172.20.128.254 (172.20.128.254)' can't be established.
ECDSA key fingerprint is SHA256:5Shvi8y7JDXfcojUkBksNtbFxmi8LtCHcn4AgzuHgXg.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '172.20.128.254' (ECDSA) to the list of known hosts.
root@172.20.128.254's password:
Linux debTP3 4.19.0-12-amd64 #1 SMP Debian 4.19.152-1 (2020-10-18) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Mon May 10 16:31:04 2021 from 172.20.128.25
root@debTP3:~#
```

Mon fichier parefeu_on actuelle

```
#Filtrage ssh
iptables -A INPUT -p tcp --dport 22 -s 172.20.128.10 -j ACCEPT
iptables -A OUTPUT -p tcp --sport 22 -d 172.20.128.10 -j ACCEPT

iptables -A INPUT  -p tcp --dport 22 -j DROP
```

Je vérifie bien que le client ne peut pas se connecter en ssh

```
root@debTP3:~# ssh root@172.20.128.254
ssh: connect to host 172.20.128.254 port 22: No route to host
root@debTP3:~#
```

Ça ne fonctionne pas

Je vais travailler à distance depuis mon client 1 DHCP je modifie mon fichier de conf pour que le ssh sois autoriser que pour mon client

Je suis connecté en ssh sur mon routeur

iptables -A OUTPUT -p icmp **–icmp-type echo-reply -j ACCEPT**

Cette commande veut dire que l’on accepte les flux icmp les envoie et reponse (ping et pong)

Je vais autoriser les requêtes icmp provenant de mes réseaux vers internet
Avant je vais remettre le nat dynamique (masquerade) pour que les requêtes passent

Comme ceci

```
#!/bin/sh
#Nat dynamique MASQUERADE
iptables -t nat -A POSTROUTING -o eth2 -j MASQUERADE
```

Autorisation icmp

```
#Autorisation ICMP
#iptables -A INPUT -p icmp --icmp-type echo-reply
#iptables -A OUTPUT -p icmp --icmp-type echo-reply
iptables -A FORWARD -i eth1 -p icmp --icmp-type echo-reply  -j ACCEPT
iptables -A FORWARD -i eth2 -p icmp --icmp-type echo-reply  -j ACCEPT
iptables -A FORWARD -i eth0 -p icmp -j ACCEPT
#iptables -A INPUT -p icmp --icmp-type echo-reply
```

Test ping client 1 vers 8.8.8.8

```
root@debTP3:~# ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=117 time=4.81 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=117 time=4.67 ms
^C
--- 8.8.8.8 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 3ms
rtt min/avg/max/mdev = 4.666/4.740/4.814/0.074 ms
root@debTP3:~#
```

Ça fonctionne parfaitement

Je vais autoriser les flux http (80), https (443), DNS(53) en gérant les différents etats

Mes règles ressemblent à sa

```
#Filtrage Http,https,dns avec les differents etats
#HTTP HTTPS
iptables -A FORWARD -p tcp --dport 80 -m state --state NEW -j ACCEPT
iptables -A FORWARD -p tcp --dport 443 -m state --state NEW -j ACCEPT
iptables -A FORWARD -p tcp --sport 80 -m state --state ESTABLISHED -j ACCEPT
iptables -A FORWARD -p tcp --sport 443 -m state --state ESTABLISHED -j ACCEPT
#DNS
iptables -A FORWARD -p udp --dport 53 -m state --state NEW -j ACCEPT
iptables -A FORWARD -p udp --sport 53 -m state --state ESTABLISHED -j ACCEPT
```

Autorisation http allant vers le port 80 en mode initié
Autorisation http venant du port 80 en mode établi

Et pareil pour https et DNS

Je test la résolution DNS avec nslookup

Je vais faire la résolution de www.google.com

Sa marche parfaitement on voit que c'est le DNS de la salle qui répond

```
root@debTP3:~# nslookup www.google.com
Server:         8.8.8.8
Address:        8.8.8.8#53

Non-authoritative answer:
Name:   www.google.com
Address: 172.217.18.196
Name:   www.google.com
Address: 2a00:1450:4007:805::2004

root@debTP3:~#
```

Je vais tester http et https j'ouvre mon navigateur

Ça ne marcher pas car pour l'état initié il faut mettre : NEW, ESTABLISHED, RELEATED l'état initié est enfaîte tous les états

Pour l'état établie nous n'autorisions que les états ESTABLISHED et RELEATED

Je vais mettre les deux états dans deux variables

Initier = NEW, ESTABLISHED, RELEATED
Établie = ESTABLISHED RELEATED

Pour éviter toute erreur bien écrire variable=valeur ne pas mettre d'espace avant ou après le « = »

Mémo = Pour le DNS nous n'avons pas besoins de définir l'état initié avec les 3 états car c'est une petite requête ça ne demande pas plusieurs trames hors que les requêtes http sa demande plusieurs trames au moins 2

Accès a Wikipédia https fonctionne correctement

Pour http sa fonctionne aussi

Lycée Jean Jacques ROUSSEAU
Montmorency

iptables -A OUTPUT -o eth0 -m state –state NEW,ESTABLISHED,RELATED -j ACCEPT

Ça veut dire que tous les paquets sortant par l'interface eth0 en etat initié sont accepter peuvent passer

iptables -A FORWARD -o eth0 -i eth1 -d 172.16.128.240/28 -p tcp –sport 80 -m state –state ESTABLISHED,RELATED -j ACCEPT passage en tcp

Tout les paquets passant par le routeur sortant de l'interface eth0 ou entrant dans eth1 a destination de 172.16.128.240/28 ayant comme source le port 80 dans l'état établie sont accepter passage en tcp

Le nat :

iptables -t nat -A PREROUTING -i eth1 -p tcp –dport 80 -j DNAT –to-destination 192.168.0.2:3128

C'est une règle de PAT qui veut dire que tous les paquet entrant sur l'interface eth1 du routeur ayant comme destination le port 80 seront rediriger vers le port 3128 de la machine qui a cette ip 192.168.0.2, en gros on a fait le mappage entre le port 80 du routeur et le port 3128 de la machine 192.168.0.2 passage en tcp

J'ai prévu le vidage de la table nat dans le script parefeuon.sh
Le masquage est déjà fait
Je vais rendre mon serveur accessible depuis l'extérieur

Je l'ai rendu accessible comme sa

```
#Filtrage ssh
iptables -t nat -A PREROUTING -i eth2 -p tcp --dport 22 -d 172.16.18.236  -j DNAT --to-destination 172.20.128.10:22
iptables -A FORWARD -i eth2 -o eth0 -d 172.20.128.10 -p tcp --dport 22 -j ACCEPT
iptables -A FORWARD  -i eth0 -o eth2 -p tcp --sport 22  -j ACCEPT
```

J’ai fait un mappage entre le port 22 de mon routeur et le port 22 de mon serveur DHCP

Et j’ai autoriser le trafic aller et retour qui passera par le routeur chaine FORWARD

Eth2 = patte du routeur dans la salle
Eth0 = Patte dans mon réseau privé

Je fais un test depuis une vm dans le réseau de la salle de classe

Je tape cette commande

```
root@DebianAdelGraphique:~# ssh root@172.16.18.236 -p 22
```

C’est bien l’ip de mon srv DHCP lorsque je fais IP a

```
root@172.16.18.236's password:
Linux srvdhcp 4.19.0-12-amd64 #1 SMP Debian 4.19.152-1 (2020-10-18) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Mon May 17 15:20:26 2021 from 172.16.18.84
root@srvdhcp:~# ip a | grep 172
    inet 172.20.128.10/24 brd 172.20.128.255 scope global eth0
root@srvdhcp:~#
```

Je vais essayer en modifiant le port de connexion en mettant le port 2000

J’ai configuré comme ceci

```
#Filtrage ssh
iptables -t nat -A PREROUTING -i eth2 -p tcp --dport 2000 -d 172.16.18.236  -j DNAT --to-destination 172.20.128.10:22
iptables -A FORWARD -i eth2 -o eth0 -d 172.20.128.10 -p tcp --dport 22 -j ACCEPT
iptables -A FORWARD  -i eth0 -o eth2 -p tcp --sport 22  -j ACCEPT
```

Sa marche parfaitement connexion ssh vers l’adresse de mon routeur sur le port 2000 me renvoie sur le port 22 de mon srv dhcp

Scan de port NMAP :

Je vais installer nmap sur mon client

nmap -s : scan les ports ouvert
nmap -sU : scan udp
nmap -sT : scan tcp

Je vais scanner mes ports ouverts avec nmap quand mon pare feu est éteint

Résultat :

```
Nmap scan report for 172.20.128.254
Host is up (0.00071s latency).
Not shown: 999 closed ports
PORT   STATE SERVICE
22/tcp open  ssh
MAC Address: 00:15:5D:D2:B5:17 (Microsoft)

Nmap done: 1 IP address (1 host up) scanned in 14.77 seconds
root@debTP3:~#
```

Lorsque j'active le pare feu sa me donne sa :

Même résultat

```
Nmap scan report for 172.20.128.254
Host is up (-0.068s latency).
Not shown: 999 filtered ports
PORT   STATE SERVICE
22/tcp open  ssh
MAC Address: 00:15:5D:D2:B5:17 (Microsoft)

Nmap done: 1 IP address (1 host up) scanned in 10.70 seconds
root@debTP3:~#
```

Protection contre scan de port avec portsentry (IPS)

Pour empêcher ce scan de port il y'a portsentry qui est très bien et qui est dans les dépôts de debian il bloquera tout scan de port sur ma machine

PortSentry est un système de détection d'intrusion) qui permet de détecter les attaques contre votre serveur dédié et particulièrement les scans de vos ports, ce système va réagir en fonction des attaques en créant dans votre pare-feu iptables de nouvelles règles de blocage.

J'installe portsentry

Fichier de configuration :

nano /etc/portsentry/portsentry.conf

Je configure comme ceci

Je met à « 1 » pour que sa bloque tout scan tcp ou udp

```
# 0 = Do not block UDP/TCP scans.
# 1 = Block UDP/TCP scans.
# 2 = Run external command only (KILL_RUN_CMD)

BLOCK_UDP="1"
BLOCK_TCP="1"
```

Je de commente cette ligne pour que n'importe quel ip qui scan mes ports soit bloquer en rajoutant une règle iptables

```
# Newer versions of Linux support the reject flag now. This
# is cleaner than the above option.
KILL_ROUTE="/sbin/route add -host $TARGET$ reject"
```

Je vais dans le fichier /etc /default /portsentry et je met sa :

```
 GNU nano 3.2                                  /etc/default/portsentry

# /etc/default/portsentry
#
# This file is read by /etc/init.d/portsentry. See the portsentry.8
# manpage for details.
#
# The options in this file refer to commandline arguments (all in lowercase)
# of portsentry. Use only one tcp and udp mode at a time.
#
TCP_MODE="atcp"
UDP_MODE="audp"
```

Portsentry va vérifier les ports utilisés et automatiquement « lier » les ports disponibles. C'est l'option la plus efficace (« a » signifie avancer). Avec cette options, portsentry établit une liste des ports d'écoute, TCP et UDP, et bloque l'hôte se connectant sur ces ports, sauf s'il est présent dans le fichier portsentry.ignore.

Lorsqu'une ip scannera mes ports elle sera détecter par portsentry qui définira une règle iptables qui bloquera les paquet qui viendront de cette machine

Cette IP sera placer dans le fichier /etc /hosts.deny c'est la liste des IP sont bloquer

Une route sera définie et pour destination l'IP mais aucune passerelle ni interface ne sera défini pour cette dernière

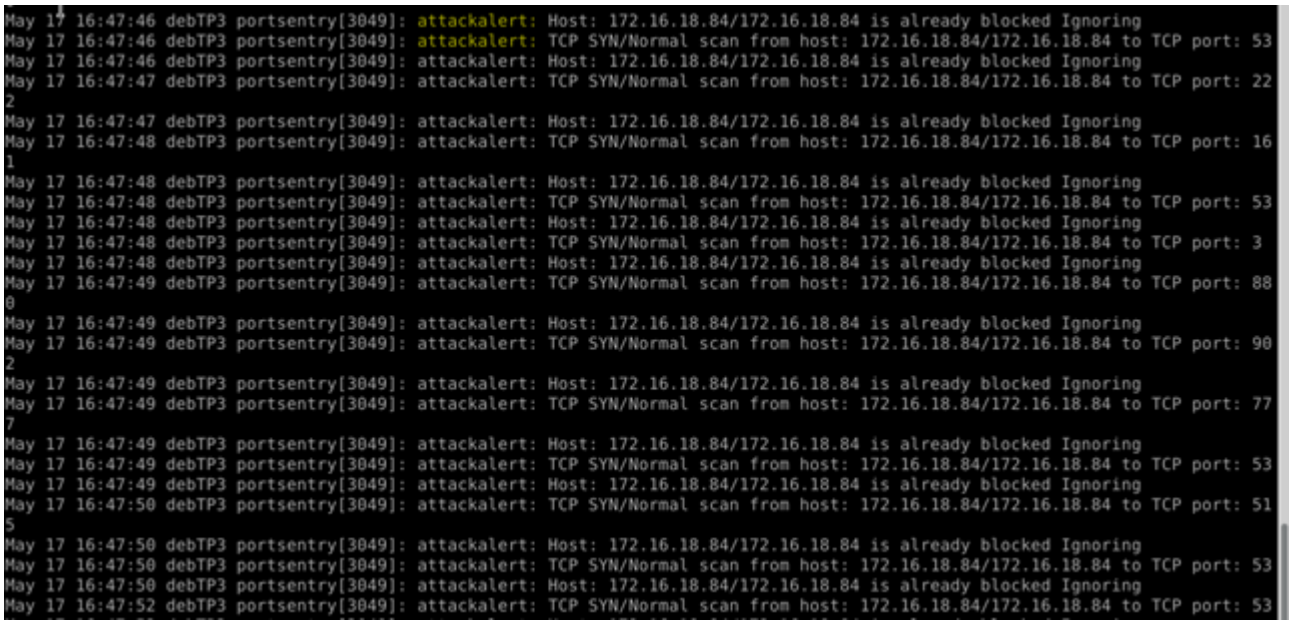
Je lance portsentry

systemctl restart networking
je procède à un scan depuis une machine

Depuis la machine qui scan :

```
Nmap scan report for 172.16.18.236
Host is up (-0.051s latency).
Not shown: 990 filtered ports
PORT     STATE  SERVICE
23/tcp   closed telnet
53/tcp   closed domain
80/tcp   closed http
139/tcp  closed netbios-ssn
143/tcp  closed imap
199/tcp  closed smux
256/tcp  closed fw1-secureremote
1025/tcp closed NFS-or-IIS
1723/tcp closed pptp
8080/tcp closed http-proxy
MAC Address: 00:15:5D:D2:B5:1C (Microsoft)
```

Dans mon fichier de log



On voit clairement « attackalert » qui m'alerte qu'un scan est en cours

Et on voit aussi que l'IP a été bloquer donc le scan ignorer

Je vérifie dans hosts.deny si l'ip y est

Elle y est

```
#            ALL EXCEPT in.fingerd: other.host.name, .other.domain
#
# If you're going to protect the portmapper use the name "rpcbind" for the
# daemon name. See rpcbind(8) and rpc.mountd(8) for further information.
#
# The PARANOID wildcard matches any host whose name does not match its
# address.
#
# You may wish to enable this to ensure any programs that don't
# validate looked up hostnames still leave understandable logs. In past
# versions of Debian this has been the default.
# ALL: PARANOID

ALL: 172.16.18.84 : DENY
ALL: 172.16.100.11 : DENY
root@debTP3:~#
```

On voit aussi qu'une route a été définis qui « isole » la machine qui m'a scanner il n'y a ni passerelle ni rien

```
root@debTP3:~# ip route show
default via 172.16.18.254 dev eth2 onlink
172.16.18.0/24 dev eth2 proto kernel scope link src 172.16.18.236
unreachable 172.16.18.84 scope host
unreachable 172.16.100.11 scope host
172.20.128.0/24 dev eth0 proto kernel scope link src 172.20.128.254
192.168.128.0/24 dev eth1 proto kernel scope link src 192.168.128.254
root@debTP3:~#
```

Je vais retirer cette machine de la liste noir en vidant mes tables iptables,vidant le fichier hosts.deny et en supprimant la route

Les deux machines peuvent recommuniquer

Je vais me faire un script qui m'envoie une alerte en cas de scan de mes ports je vais utiliser le paquets swaks qui permet d'envoyer des mails en ligne de commande

Je crée un script qui analyse le fichier de log et si il y'a le mot «attackalert» m'enverra le fichier log épurer en affichant que les lignes qui contiennent attackAlert

Je vérifie si dans le fichier le mot attackalert est présent

Si il est présent toutes les phrases contenant ce mot seront envoyer dans le fichier logEnvoie.tkt

Ce dernier me sera envoyé par mail en ligne de commande automatiquement

Je fais un test je vérifie sur mon mail si sa a bien été envoyer

Le fichier m'a bien été envoyer

```
#!/bin/sh
cat /var/log/syslog | grep attackalert
resultat=$?
echo "$resultat"
if [ $resultat = 0 ]
then
cat /var/log/syslog | grep attackalert > logEnvoie.txt
        swaks -t adel.sadek752@gmail.com -s smtp.gmail.com 587 -tls -au alerteport942@gmail.com -ap          --h-Subject "Alerte scan de port détecter sur votre machine Debian" --body "Un$
fi
```

Je vais maintenant intégrer ce script dans cron qui l'exécutera toutes les minutes

Exécution toute les minutes du script portMail.sh

```
#
# m h  dom mon dow   command
 */1 * * * * /root/port/portMail.sh
```

Sa marche parfaitement !

Je modifie mon fichier pour qu'il vide le fichier de log après l'envoie

Rajout fail2ban

J'installe le paquet fail2ban

Il y'a deux fichiers de configuration dans /etc /fail2ban

fail2ban.conf qui n'a pas besoin d'être modifier
Et jail.conf c'est là ou on précise les services à surveiller et le log à surveiller ainsi que le mot clé à rechercher

Il y'a une directive [DEFAULT] ou tout ce qui est mentionner dedans sera appliquer à toute les différentes jails ont pourra y préciser
bantime (le temps de bannissement d'une ip)
maxretry (au bout de combien de tentatives on exécute l'action de bannissement)
logpath (dans quel log chercher)
filter (le mot clé à chercher dans le log)

Ici pour ssh la jail ressemble à sa :

```
[sshd]

# To use more aggressive sshd modes set filter parameter "mode" in jail.local:
# normal (default), ddos, extra or aggressive (combines all).
# See "tests/files/logs/sshd" or "filter.d/sshd.conf" for usage example and details.
#mode   = normal
port    = 22
logpath = /var/log/auth.log
filter = sshd
```

J'ai créé un utilisateur fail avec comme mdp fail et j'ai essayé de me connecter plus de 5 fois en ssh avec un mdp erroner

Au bout de 5 tentatives sa m'a afficher ceci

```
root@debTP3:/etc/fail2ban# ssh fail@172.16.18.236
ssh: connect to host 172.16.18.236 port 22: Connection refused
```

Ont voient bien qu'après 5 tentatives l'IP a été ban ont voient ceci dans *//var /log /fail2ban.log*

Je vais regarder dans mes règles iptables si sa a bien été configurer

```
Chain f2b-sshd (1 references)
target     prot opt source               destination
REJECT     all  --  172.16.18.236        anywhere             reject-with icmp-port-unreachable
RETURN     all  --  anywhere             anywhere
root@debTP3:/etc/fail2ban#
```

Ont voient clairement l'IP qui a été banni

Si je modifie dans jail.conf le port de ssh et que je modifie le port de ssh dans son fichier de conf par le numéro 1025 ça fonctionne parfaitement

Je vais vérifier si pour sftp c'est pareil ou il faut refaire une autre directive

Sa marche parfaitement avec sftp

```
2021-05-20 10:18:36,486 fail2ban.filter        [1313]: INFO    [sshd] Found 172.16.18.236 - 2021-05-20 10:18:36
2021-05-20 10:18:38,215 fail2ban.filter        [1313]: INFO    [sshd] Found 172.16.18.236 - 2021-05-20 10:18:38
2021-05-20 10:18:38,528 fail2ban.actions       [1313]: NOTICE  [sshd] Ban 172.16.18.236
```

Ça bloque l'ip directement comme ssh

Ici depuis un client Windows accès depuis Filezilla

```
2021-05-20 10:54:26,201 fail2ban.filter        [2019]: INFO    [sshd] Found 172.16.18.18 - 2021-05
-20 10:54:26
2021-05-20 10:54:27,959 fail2ban.filter        [2019]: INFO    [sshd] Found 172.16.18.18 - 2021-05
-20 10:54:27
2021-05-20 10:54:28,593 fail2ban.actions       [2019]: NOTICE  [sshd] Ban 172.16.18.18
root@debTP3:/etc/fail2ban#
```

Lorsque je supprime la règle iptables pour ré-autoriser une adresse IP ça ne rebanni pas lorsque je me connecte 2 fois avec un mdp erroné

```
2021-05-20 10:57:48,950 fail2ban.actions       [2019]: WARNING [sshd] 172.16.18.18 already banned
2021-05-20 10:57:53,630 fail2ban.filter        [2019]: INFO    [sshd] Found 172.16.18.18 - 2021-05-20 10:57:53
```

Il me dit que l'adresse IP est déjà bannie donc il ne la rebannie pas

Mémo : Pour ssh si je veux autoriser un utilisateur que pour sftp et non ssh il faut que je lui précise comme shell « / usr /sbin /nologin » qui interdira toute connexion via ssh

Par exemple pour l'utilisateur fail

```
root@debTP3:/# usermod fail -s /usr/sbin/nologin
root@debTP3:/# ssh fail@172.16.18.236
ssh: connect to host 172.16.18.236 port 22: Connection refused
root@debTP3:/# ssh fail@172.16.18.236 -p 2025
fail@172.16.18.236's password:
Linux debTP3 4.19.0-12-amd64 #1 SMP Debian 4.19.152-1 (2020-10-18) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Thu May 20 10:11:36 2021 from 172.16.18.236
This account is currently not available.
Connection to 172.16.18.236 closed.
root@debTP3:/#
```

Mais quand j'essaye sftp via Filezilla sa marche parfaitement

Donc voilà comment restreindre l'accés à ssh en autorisant sftp

Le shell de l'utilisateur root c'est /bin / bash
Celui qui interdit l'accés ssh est /usr / sbin /nologin