TP 6 - Routage, filtrage et NAT

Cablage et configuration du réseau local

Configuration

Dans la partie interface de marionnet, on définit en dur les adresses ip de chaque machine qu'on utilise dans ce tp. Les adresses des machines sont les suivantes :

machine	reseau Local	adresse
1	LAN_1	192.168.1.1/24
2	LAN_1	192.168.1.2/24
3	LAN_2	10.0.0.3/8
4	LAN_2	10.0.0.4/8
routeur	LAN_1	192.168.1.254/24
routeur	LAN_2	10.255.255.254/8
routeur	CAN_1	145.12.0.53/16
intrus	CAN_1	145.12.0.42/16

Sur la machine *routeur*, on autorise la passerelle, en tapant la commande suivante :

```
echo 1 > /proc/sys/net/ipv4/ip_forward
```

Sur les machines de LAN_1, on utilise la commande suivante pour utliser le routeur comme passerelle :

```
route add -net default gw 192.168.1.254
```

Sur les machines de LAN_2, on utilise la commande suivante pour utliser le routeur comme passerelle :

```
route add -net default gw 10.255.255.254
```

Sur les machines de CAN_1, on utilise la commande suivante pour utliser le routeur comme passerelle :

```
route add -net default gw 145.12.0.53
```

Test

- test ping *m1* $\rightarrow$ *intrus* :

```
 ping 145.12.0.42
PING 145.12.0.42 (145.12.0.42) 56(84) bytes of data.
64 bytes from 145.12.0.42: icmp_seq=1 ttl=63 time=14.6 ms
64 bytes from 145.12.0.42: icmp_seq=2 ttl=63 time=2.68 ms
64 bytes from 145.12.0.42: icmp_seq=3 ttl=63 time=2.48 ms
64 bytes from 145.12.0.42: icmp_seq=4 ttl=63 time=2.94 ms
64 bytes from 145.12.0.42: icmp_seq=5 ttl=63 time=2.81 ms
```

- capture des trames de tcpdump sur *intrus* :

```
13:22:28.590864 arp who-has 145.12.0.42 tell 145.12.0.53
13:22:28.590917 arp reply 145.12.0.42 is-at 02:04:06:d9:77:54 (oui Unknown)
13:22:28.591827 IP 192.168.1.1 > 145.12.0.42: ICMP echo request, id 47112, seq 1, length 64
13:22:28.591887 IP 145.12.0.42 > 192.168.1.1: ICMP echo reply, id 47112, seq 1, length 64
13:22:28.599265 IP 145.12.0.42.3075 > 85.37.17.55.domain: 3375+ PTR? 42.0.12.145.in-addr.arpa.
(42)
13:22:29.588377 IP 192.168.1.1 > 145.12.0.42: ICMP echo request, id 47112, eq 2, length 64
13:22:29.588432 IP 145.12.0.42 > 192.168.1.1: ICMP echo reply, id 47112, seq 2, length 64
13:22:30.604036 IP 192.168.1.1 > 145.12.0.42: ICMP echo request, id 47112, seq 3, length 64
13:22:30.604087 IP 145.12.0.42 > 192.168.1.1: ICMP echo reply, id 47112, seq 3, length 64
13:22:31.608266 IP 145.12.0.53 > 145.12.0.42: ICMP host 85.37.17.55 unreachable, length 78
13:22:31.610524 IP 192.168.1.1 > 145.12.0.42: ICMP echo request, id 47112, seq 4, length 64
13:22:31.610575 IP 145.12.0.42 > 192.168.1.1: ICMP echo reply, id 47112, seq 4, length 64
13:22:32.617744 IP 192.168.1.1 > 145.12.0.42: ICMP echo request, id 47112, seq 5, length 64
13:22:32.617798 IP 145.12.0.42 > 192.168.1.1: ICMP echo reply, id 47112, seq 5, length 64
```

Filtrage & SNAT

Toutes les commandes décrites dans cette section sont réalisées sur la machine *routeur* :

1. Pour bloquer toutes les données entrant dans le réseau à partir de l'*intrus*, on utilise les commandes suivantes :

```
iptables -N blockEth2
iptables -A blockEth2 -i eth2 -j DROP
iptables -A FORWARD -j blockEth2
```

2. Ensuite on autorise l'*intrus* à communiquer seulement avec les machines déjà connectées

```
iptables -A blockEth2 -m state --state ESTABLISHED -j ACCEPT
```

capture des trames :

- *m1* → *intrus* en utilisant *tcpdump* sur *intrus* :

```
22:30:22 IP 192.168.1.1 > 145.12.0.42: ICMP echo request,id 22027, seq 1,length 64
22:30:22 IP 145.12.0.42 > 192.168.1.1: ICMP echo reply, id 22027,  seq 1,length 64
22:30:23 IP 192.168.1.1 > 145.12.0.42: ICMP echo request, id 22027, seq 2, length 64
22:30:23 IP 145.12.0.42 > 192.168.1.1: ICMP echo reply, id 22027, seq 2, length 64
22:30:24 IP 192.168.1.1 > 145.12.0.42: ICMP echo request, id 22027, seq 3, length 64
22:30:24 IP 145.12.0.42 > 192.168.1.1: ICMP echo reply, id 22027, seq 3, length 64
22:30:25 IP 192.168.1.1 > 145.12.0.42: ICMP echo request, id 22027, seq 4, length 64
22:30:25 IP 145.12.0.42 > 192.168.1.1: ICMP echo reply, id 22027, seq 4, length 64
22:30:26 IP 192.168.1.1 > 145.12.0.42: ICMP echo request, id 22027, seq 5, length 64
22:30:26 IP 145.12.0.42 > 192.168.1.1: ICMP echo reply, id 22027, seq 5, length 64
22:30:27 arp who-has 145.12.0.42 tell 145.12.0.53
22:30:27 arp reply 145.12.0.42 is-at 02:04:06:6e:10:d5 (oui Unknown)
22:30:27 IP 192.168.1.1 > 145.12.0.42: ICMP echo request, id 22027, seq 6, length 64
22:30:27 IP 145.12.0.42 > 192.168.1.1: ICMP echo reply, id 22027, seq 6, length 64
22:30:28 IP 192.168.1.1 > 145.12.0.42: ICMP echo request, id 22027, seq 7, length 64
22:30:28 IP 145.12.0.42 > 192.168.1.1: ICMP echo reply, id 22027, seq 7, length 64
```

- *m1* → *intrus* en utilisant *tcpdump* sur *m2* :

```
22:30:22 IP 192.168.1.1 > 145.12.0.42: ICMP echo request, id 22027, seq 1, length 64
22:30:23 IP 192.168.1.1 > 145.12.0.42: ICMP echo request, id 22027, seq 2, length 64
22:30:24 IP 192.168.1.1 > 145.12.0.42: ICMP echo request, id 22027, seq 3, length 64
22:30:25 IP 192.168.1.1 > 145.12.0.42: ICMP echo request, id 22027, seq 4, length 64
22:30:26 IP 192.168.1.1 > 145.12.0.42: ICMP echo request, id 22027, seq 5, length 64
22:30:27 arp who-has 192.168.1.254 tell 192.168.1.1
22:30:27 arp reply 192.168.1.254 is-at 02:04:06:3b:03:04 (oui Unknown)
22:30:27 IP 192.168.1.1 > 145.12.0.42: ICMP echo request, id 22027, seq 6, length 64
22:30:28 IP 192.168.1.1 > 145.12.0.42: ICMP echo request, id 22027, seq 7, length 64
```

On ne voit que les requêtes du protocole *ICMP* sur *m2*, ce qui est tout à fait logique car avec les chaines *iptables* que nous utilisons, les paquets ne peuvent être echangés qu'avec les machine déjà connectées.

3. On modifie par la suite les paramètres, de sorte à ce que l'*intrus* aura toujours l'impression de communiquer avec le *routeur*

```
1  iptables -t nat -A POSTROUTING -o eth2 -j SNAT --to 145.12.0.53
```

- capture de trames sur *intrus* en utilisant *tcpdump:*

  m1 → *intrus* :

```
22:43:03 IP 145.12.0.53 > 145.12.0.42: ICMP echo request, id 35851, seq 1, length 64
22:43:24 IP 145.12.0.42 > 145.12.0.53: ICMP echo reply, id 35851, seq 1, length 64
22:43:04 IP 145.12.0.53 > 145.12.0.42: ICMP echo request, id 35851, seq 2, length 64
22:43:04 IP 145.12.0.42 > 145.12.0.53: ICMP echo reply, id 35851, seq 2, length 64
22:43:05 IP 145.12.0.53 > 145.12.0.42: ICMP echo request, id 35851, seq 3, length 64
22:43:05 IP 145.12.0.42 > 145.12.0.53: ICMP echo reply, id 35851, seq 3, length 64
22:43:06 IP 145.12.0.53 > 145.12.0.42: ICMP echo request, id 35851, seq 4, length 64
22:43:06 IP 145.12.0.42 > 145.12.0.53: ICMP echo reply, id 35851, seq 4, length 64
22:43:07 IP 145.12.0.53 > 145.12.0.42: ICMP echo request, id 35851, seq 5, length 64
22:43:07 IP 145.12.0.42 > 145.12.0.53: ICMP echo reply, id 35851, seq 5, length 64
22:43:08 IP 145.12.0.53 > 145.12.0.42: ICMP echo request, id 35851, seq 6, length 64
22:43:08 IP 145.12.0.42 > 145.12.0.53: ICMP echo reply, id 35851, seq 6, length 64
22:43:09 IP 145.12.0.53 > 145.12.0.42: ICMP net 10.10.0.10 unreachable, length 78
22:43:09 IP 145.12.0.53 > 145.12.0.42: ICMP echo request, id 35851, seq 7, length 64
22:43:09 IP 145.12.0.42 > 145.12.0.53: ICMP echo reply, id 35851, seq 7, length 64
```