TP6

1 Routage

1.1 Préliminaires

Première machine `ermengaud` :

Avec la commande `/sbin/route -n` :

```
Table de routage IP du noyau
Destination     Passerelle      Genmask         Indic Metric Ref    Use Iface
0.0.0.0         10.0.202.254    0.0.0.0         UG    0      0        0 eth0
10.0.202.0      0.0.0.0         255.255.255.0   U     0      0        0 eth0
```

Avec la commande `ip route ls` :

```
default via 10.0.202.254 dev eth0 onlink
10.0.202.0/24 dev eth0 proto kernel scope link src 10.0.202.7
```

Deuxième machine `fabre` :

Avec la commande `/sbin/route -n` :

```
Table de routage IP du noyau
Destination     Passerelle      Genmask         Indic Metric Ref    Use Iface
0.0.0.0         10.0.202.254    0.0.0.0         UG    0      0        0 eth0
10.0.202.0      0.0.0.0         255.255.255.0   U     0      0        0 eth0
```

Avec la commande `ip route ls`

```
default via 10.0.202.254 dev eth0 onlink
10.0.202.0/24 dev eth0 proto kernel scope link src 10.0.202.8
```

- Les deux machines sont dans le même réseau local (10.0.202.0/24 Ligne 2 des commandes).
- L'adresse de la passerelle de la route par défaut est `10.0.202.254`.
- Le suffixe `24` correspond au `Genmask` qui est le masque du réseau de destination (`24` correspond aux 24 premiers bits ce qui donne le masque `255.255.255.0`).

> en IPv6

Avec la commande `ip -6 route ls` :

Première machine `ermengaud` :

```
2001:660:6101:800:202::/80 dev eth0 proto kernel metric 256  pref medium
fe80::/64 dev eth0 proto kernel metric 256  pref medium
fe80::/64 dev br0 proto kernel metric 256  pref medium
default via fe80::5a20:b1ff:feb1:2300 dev eth0 proto ra metric 1024  expires
8613sec hoplimit 25 pref medium
```

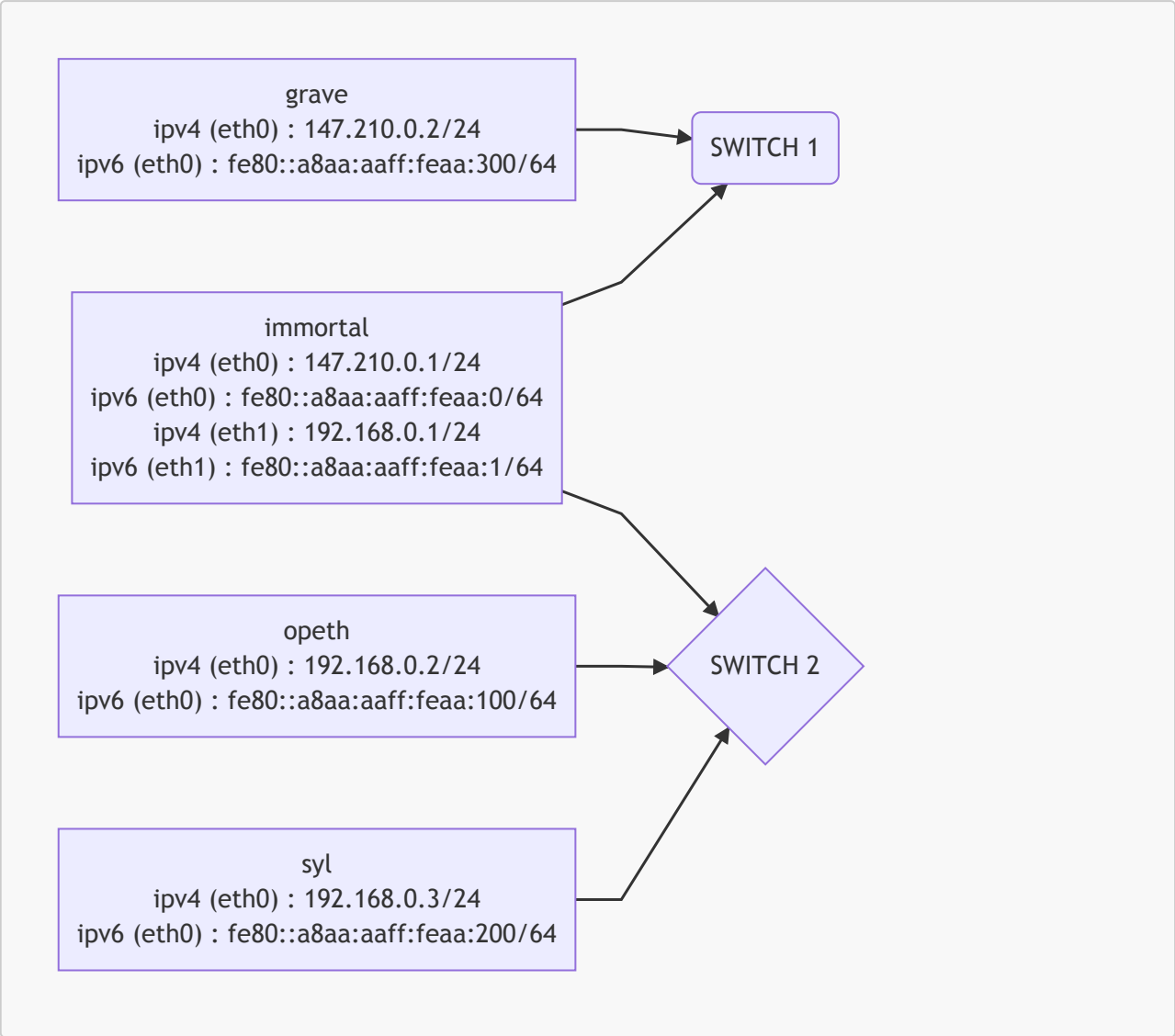
Deuxième machine `fabre` :

```
2001:660:6101:800:202::/80 dev eth0 proto kernel metric 256  pref medium
fe80::/64 dev eth0 proto kernel metric 256  pref medium
fe80::/64 dev br0 proto kernel metric 256  pref medium
default via fe80::5a20:b1ff:feb1:2300 dev eth0 proto ra metric 1024  expires
8965sec hoplimit 25 pref medium
```

- La partie réseau de l'adresse IPv6 `2001:660:6101:800:202::/80` est `2001:660:6101:800:202` (ce sont les `5` premiers paquets séparés par `:` ; les `17` premiers caractères hexadécimaux sans le 0 non-significatif ou les `20` avec les 0). La partie hôte est sur `3` paquets (48 bits).
- ✓.
- ✓ c'est `fe80::5a20:b1ff:feb1:2300`.

1.2 Routage Basique

- Les adresses IP des machines du réseau :



- Les machines peuvent communiquer dans leurs réseaux locaux respectifs :
  - `grave` peut ping `immortal` (sur eth0) et inversement.
  - `immortal` (sur eth1), `opeth` et `syl` peuvent se ping entre-elles dans les deux sens.
- Les tables de routage des machines, que j'obtiens avec la commande `route -n` sont :

`grave` :

```
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
147.210.0.0     0.0.0.0         255.255.255.0   U     0      0        0 eth0
```

`immortal` :

```
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
147.210.0.0     0.0.0.0         255.255.255.0   U     0      0        0 eth0
192.168.0.0     0.0.0.0         255.255.255.0   U     0      0        0 eth1
```

`opeth` et `syl` :

```
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
192.168.0.0     0.0.0.0         255.255.255.0   U     0      0        0 eth0
```

- J'ai configuré la table de routage de `grave` avec la commande `route add default gw 147.210.0.1` pour que par défaut les paquets qui ne sont pas destinés à un réseau dans la table de routage, passent par `immortal`. J'ai aussi configuré les tables de routage de `opeth` et `syl` avec la commande `route add default gw 192.168.0.1` pour la même raison. Et j'ai activé le relais des paquets sur `immortal` avec la commande `echo 1 > /proc/sys/net/ipv4/ip_forward`. Pour finir j'ai testé que toutes les machines sont capables de communiquer ensemble et c'est le cas.

- Lorsque j'envoie un ping de `opeth` vers `grave` avec la commande `ping 147.210.0.2 -c 1` je peux voir que les paquets passent bien par `immortal` je peux aussi voir les requêtes arp de `immortal` sur les 2 réseaux pour faire la résolution ip.

  Avec la commande `tcpdump -n -i any` :

```
14:26:58.018402 IP 192.168.0.2 > 147.210.0.2: ICMP echo request, id 772, seq 1,
length 64
14:26:58.018431 IP 192.168.0.2 > 147.210.0.2: ICMP echo request, id 772, seq 1,
length 64
14:26:58.020529 IP 147.210.0.2 > 192.168.0.2: ICMP echo reply, id 772, seq 1,
length 64
14:26:58.020537 IP 147.210.0.2 > 192.168.0.2: ICMP echo reply, id 772, seq 1,
length 64
14:27:03.024756 ARP, Request who-has 192.168.0.1 tell 192.168.0.2, length 46
14:27:03.024793 ARP, Reply 192.168.0.1 is-at aa:aa:aa:aa:00:01, length 28
14:27:03.025112 ARP, Request who-has 147.210.0.1 tell 147.210.0.2, length 46
14:27:03.025123 ARP, Reply 147.210.0.1 is-at aa:aa:aa:aa:00:00, length 28
```

- Je l'ai déjà fait.

1.3 Routage Avancé (Bonus)

Je configure la table de routage de `opeth` et de toutes les machines du sous-réseau `147.210.12.0/24`, hormis la passerelle, avec la commande `route add default gw 147.210.12.2` pour que la passerelle par défaut soit `immortal`.

Je configure la table de routage de `immortal` et de toutes les machines du sous-réseau `147.210.13.0/24`, hormis la passerelle, avec la commande `route add default gw 147.210.13.2` pour que la passerelle par défaut soit `grave`.

Je configure la table de routage de `syl` et de toutes les machines du sous-réseau `147.210.14.0/24`, hormis la passerelle, avec la commande `route add default gw 147.210.14.1` pour que la passerelle par défaut soit `grave`.

Je configure la table de routage de `nile` et de toutes les machines du sous-réseau `147.210.15.0/24`, hormis la passerelle, avec la commande `route add default gw 147.210.15.1` pour que la passerelle par défaut soit `syl`.

J'active le relais des paquets sur les 3 passerelles qui sont `immortal`, `grave` et `syl` avec la commande `echo 1 > /proc/sys/net/ipv4/ip_forward`.

Pour finir, j'ajoute 4 routes à `grave` :

- La première pour que les machines du sous-réseau `147.210.12.0/24` puisse communiquer avec celles du sous-réseau `147.210.15.0/24` et pour que celles du sous-réseau `147.210.13.0/24` puisse répondre à celles du sous-réseau `147.210.15.0/24`. J'utilise la commande `route add -net 147.210.15.0/24 gw 147.210.14.2` pour ajouter la route.
- La deuxième pour que les machines du sous-réseau `147.210.15.0/24` puisse communiquer avec celles du sous-réseau `147.210.12.0/24` et pour que celles du sous-réseau `147.210.14.0/24` puisse répondre à celles du sous-réseau `147.210.12.0/24`. J'utilise la commande `route add -net 147.210.12.0/24 gw 147.210.13.1` pour ajouter la route.

1 Firewall

- La différence entre la DMZ (zone démilitarisée) et le réseau interne des employés est que les machines qui sont dans la DMZ sont accessibles depuis Internet, mais pas les machines des employés.
- C'est fait ✓.
- Effectivement plus aucun trafic réseau n'est autorisé vers ou à travers `immortal` :

```
root@opeth:~#  ping 147.210.0.1 -c 4
PING 147.210.0.1 (147.210.0.1) 56(84) bytes of data.

--- 147.210.0.1 ping statistics ---
4 packets transmitted, 0 received, 100% packet loss, time 3023ms

root@opeth:~#  ping 192.168.0.2 -c 4
PING 192.168.0.2 (192.168.0.2) 56(84) bytes of data.

--- 192.168.0.2 ping statistics ---
4 packets transmitted, 0 received, 100% packet loss, time 3025ms

root@opeth:~#  ping 192.168.1.2 -c 4
PING 192.168.1.2 (192.168.1.2) 56(84) bytes of data.

--- 192.168.1.2 ping statistics ---
4 packets transmitted, 0 received, 100% packet loss, time 3000ms
```

- Pour autoriser le ping du réseau Interne vers Internet, je fais ces 2 commandes :
    - `iptables -A FORWARD -i eth2 -o eth0 -p icmp -j ACCEPT`
    - `iptables -A FORWARD -i eth2 -o eth1 -p icmp -j ACCEPT`
- Je fais le test avec la commande `ping 147.210.0.2 -c 4` sur `nile` :

```
PING 147.210.0.2 (147.210.0.2) 56(84) bytes of data.

--- 147.210.0.2 ping statistics ---
4 packets transmitted, 0 received, 100% packet loss, time 3000ms
```

Je tape donc la commande `iptables -A FORWARD -m state --state RELATED,ESTABLISHED -j ACCEPT` sur `imortal`.

Après avoir tapé la commande on remarque que les ping fonctionnent du réseau interne à internet mais pas l'inverse :

```
PING 147.210.0.2 (147.210.0.2) 56(84) bytes of data.
64 bytes from 147.210.0.2: icmp_seq=1 ttl=63 time=0.887 ms
64 bytes from 147.210.0.2: icmp_seq=2 ttl=63 time=0.736 ms
64 bytes from 147.210.0.2: icmp_seq=3 ttl=63 time=0.710 ms
64 bytes from 147.210.0.2: icmp_seq=4 ttl=63 time=0.578 ms

--- 147.210.0.2 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3003ms
rtt min/avg/max/mdev = 0.578/0.727/0.887/0.109 ms
```

- Pour autoriser l'accès au web depuis les machines du réseau interne j'utilise la commande `iptables -A FORWARD -i eth2 -o eth0 -p tcp --dport 80 -j ACCEPT` et la commande `iptables -A FORWARD -i eth2 -o eth3 -p tcp --dport 80 -j ACCEPT`. Je teste avec `wget` :

```
root@nile:~# wget 147.210.0.2
--2020-04-19 20:34:23--  http://147.210.0.2/
Connecting to 147.210.0.2:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 10701 (10K) [text/html]
Saving to: ‘index.html’

index.html          100%[===================>]  10.45K  --.-KB/s    in 0s

2020-04-19 20:34:23 (99.4 MB/s) - ‘index.html’ saved [10701/10701]
```

```
root@nile:~# wget 172.16.0.2
--2020-04-19 20:36:20--  http://172.16.0.2/
Connecting to 172.16.0.2:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 10701 (10K) [text/html]
Saving to: ‘index.html.1’

index.html.1        100%[===================>]  10.45K  --.-KB/s    in 0s

2020-04-19 20:36:20 (122 MB/s) - ‘index.html.1’ saved [10701/10701]
```

- Pour autoriser `grave` à accéder au serveur ssh de `dt` je tape la commande `iptables -A FORWARD -s 172.16.0.2 -d 192.168.0.3 -p tcp --dport 22 -j ACCEPT`.

Je teste avec le compte toto (`ssh toto@192.168.0.3`) :

Sur `grave` :

```
root@grave:~# ssh toto@192.168.0.3
The authenticity of host '192.168.0.3 (192.168.0.3)' can't be established.
ECDSA key fingerprint is SHA256:b2tuLYwJkZtgLmH5GkvZyi2JWc/v8plfeyPmuz9cxmU.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.0.3' (ECDSA) to the list of known hosts.
toto@192.168.0.3's password:
Linux dt 4.7.0-1-amd64 #1 SMP Debian 4.7.2-1 (2016-08-28) 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.
toto@dt:~$
```

Sur `opeth` et sur `nile` :

```
root@opeth:~# ssh toto@192.168.0.3
ssh: connect to host 192.168.0.3 port 22: Connection timed out
```

```
root@nile:~#  ssh toto@192.168.0.3
ssh: connect to host 192.168.0.3 port 22: Connection timed out
```

- Pour autoriser l'accès depuis n'importe où vers le serveur web de `syl` j'utilise la commande `iptables -A FORWARD -s 0.0.0.0 -d 192.168.0.2 -p tcp --dport 80 -j ACCEPT`. Je teste avec `wget` :

Test avec une machine du réseau interne :

```
root@nile:~# wget 192.168.0.2
--2020-04-19 21:01:24--  http://192.168.0.2/
Connecting to 192.168.0.2:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 10701 (10K) [text/html]
Saving to: ‘index.html.2’

index.html.2        100%[===================>]  10.45K  --.-KB/s    in 0s

2020-04-19 21:01:24 (113 MB/s) - ‘index.html.2’ saved [10701/10701]
```

Test avec une machine du réseau externe :

```
root@opeth:~#  wget 192.168.0.2
--2020-04-19 21:03:14--  http://192.168.0.2/
Connecting to 192.168.0.2:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 10701 (10K) [text/html]
Saving to: ‘index.html’

index.html          100%[===================>]  10.45K  --.-KB/s    in 0s

2020-04-19 21:03:14 (192 MB/s) - ‘index.html’ saved [10701/10701]
```

Test avec une machine de la DMZ :

```
root@dt:~#  wget 192.168.0.2
--2020-04-19 21:05:35--  http://192.168.0.2/
Connecting to 192.168.0.2:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 10701 (10K) [text/html]
Saving to: ‘index.html’

index.html          100%[===================>]  10.45K  --.-KB/s    in 0s

2020-04-19 21:05:35 (187 MB/s) - ‘index.html’ saved [10701/10701]
```

- Je teste le firewall sur la DMZ avec `nmap` depuis les machines `opeth` et `grave` :

Sur `grave` je constate que j'ai accès au port 80 en tcp sur le serveur `syl` et que j'ai accès au port 22 en tcp du serveur `dt` :

```
root@grave:~# nmap -Pn -F 192.168.0.2

Starting Nmap 7.12 ( https://nmap.org ) at 2020-04-19 21:21 UTC
Nmap scan report for 192.168.0.2
Host is up (0.0014s latency).
Not shown: 99 filtered ports
PORT   STATE SERVICE
80/tcp open  http

Nmap done: 1 IP address (1 host up) scanned in 3.86 seconds
```

```
root@grave:~#  nmap -Pn -F 192.168.0.3

Starting Nmap 7.12 ( https://nmap.org ) at 2020-04-19 21:21 UTC
Nmap scan report for 192.168.0.3
Host is up (0.00069s latency).
Not shown: 99 filtered ports
PORT   STATE SERVICE
22/tcp open  ssh

Nmap done: 1 IP address (1 host up) scanned in 5.65 seconds
```

Sur `opeth` je constate que j'ai également accès au port 80 en tcp sur le serveur `syl` et que je n'ai accès à aucun des ports du serveur `dt` :

```
root@opeth:~# nmap -Pn -F 192.168.0.2

Starting Nmap 7.12 ( https://nmap.org ) at 2020-04-19 21:26 UTC
Nmap scan report for 192.168.0.2
Host is up (0.00099s latency).
Not shown: 99 filtered ports
PORT   STATE SERVICE
80/tcp open  http

Nmap done: 1 IP address (1 host up) scanned in 5.65 seconds
```

```
root@opeth:~# nmap -Pn -F 192.168.0.3

Starting Nmap 7.12 ( https://nmap.org ) at 2020-04-19 21:28 UTC
Nmap scan report for 192.168.0.3
Host is up.
All 100 scanned ports on 192.168.0.3 are filtered

Nmap done: 1 IP address (1 host up) scanned in 21.07 seconds
```