TP 4 Compte rendu Fabien Mauhourat

## Configuration de base

**Configuration des interfaces réseau :**

- enp0s3 → nat
- enp0s8 → réseau privé

sudo vim /etc/netplan/50-cloud-init.yaml :

```
network:
  ethernets:
    enp0s3:
      dhcp4: true
    enp0s8:
      dhcp4: true
  version: 2
```

**Appliquer la configuration :**

sudo netplan apply

**Vérifier l'attribution des IPs par le DHCP :**

ip addr show | grep enp

```
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP
group default qlen 1000
  inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic enp0s3
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP
group default qlen 1000
  inet 192.168.59.201/24 brd 192.168.59.255 scope global dynamic enp0s8
```

**Installation de qemu et lxc :**

sudo apt install qemu lxc lxc-templates debian-archive-keyring

**Installation des services et paquets de la machine alpine :**

apk update && apk add openssh-client openssh-server iproute2 iputils bind-tools bash-completion
service sshd start
rc-update add sshd

**Autarisation des bridges dans qemu :**

sudo mkdir -p /etc/qemu
sudo vim /etc/qemu/bridge.conf :
allow br0
allow br1

**Création des clones :**

```
for i in {1..2..1}; do \
        qemu-img create -f qcow2 -b alpine.qcow2 alpine_tp4_${i}.qcow2 \
done
```

**Activation de l'ip forward :**

- Non persistent : sudo sysctl net.ipv4.ip_forward=1
- Persistent : sudo bash -c 'echo "net.ipv4.ip_forward=1" >> /etc/sysctl.conf'

## Bridge au démarrage

**Configuration réseau privé hôte (private bridge)**

Plan d'adressage : 192.168.50.0/24

**Question 1**

**sudo vim /etc/netplan/50-cloud-init.yaml :**

```
network:
   bridges:
      br0:
         addresses: [ 192.168.50.1/24 ]
```

sudo netplan apply

**Donner une adresse ip au bridge :**

sudo ip addr add 192.168.50.1/24 broadcast 192.168.50.255 dev br0

**Question 2**

sudo qemu-system-x86_64 -hda alpine_tp4_1.qcow2 -netdev bridge,id=net0,br=br0 -device e1000,netdev=net0,mac=52:54:00:12:34:55 -k fr -m 128M -vnc :0 &

**Configuration sur l'invité 1 :**

ip link set eth0 up
ip addr add 192.168.50.2/24 broadcast 192.168.50.255 dev eth0
ip route add default via 192.168.50.1

sudo qemu-system-x86_64 -hda alpine_tp4_2.qcow2 -netdev bridge,id=net0,br=br0 -device e1000,netdev=net0,mac=52:54:00:12:34:56 -k fr -m 128M -vnc :0 &

**Configuration sur l'invité 2 :**

ip link set eth0 up
ip addr add 192.168.50.3/24 broadcast 192.168.50.255 dev eth0
ip route add default via 192.168.50.1

**Redirection iptables pour ssh :**

sudo iptables -t nat -A PREROUTING -i enp0s8 -p tcp --dport 10022 -j DNAT --to 192.168.50.2:22
sudo iptables -t nat -A POSTROUTING -o enp0s8 -p tcp --dport 10022 -d 192.168.50.2 -j SNAT --to 192.168.59.201

**Question 4**

lxc-create -n debian_1 -t debian -- --packages=iproute2,dnsutils,iputils-ping
lxc-create -n debian_2 -t debian -- --packages=iproute2,dnsutils,iputils-ping

sed -i 's/lxcbr0/br0/g' /var/lib/lxc/debian_1/config /var/lib/lxc/debian_2/config

sed -i '/lxc.net.0.hwaddr*/a lxc.net.0.ipv4.address = 192.168.50.4/24' /var/lib/lxc/debian_1/config

sed -i '/lxc.net.0.hwaddr*/a lxc.net.0.ipv4.address = 192.168.50.5/24' /var/lib/lxc/debian_2/config

sed -i '/lxc.net.0.ipv4*/a lxc.net.0.ipv4.gateway = 192.168.50.1' /var/lib/lxc/debian_2/config /var/lib/lxc/debian_2/config

**Debian 1 :**
Mode : satic
IP : 192.168.50.4
Gateway : 192.168.50.1

**Debian 2 :**
Mode : satic
IP : 192.168.50.5
Gateway : 192.168.50.1

**Question 3,5,6**

**Communications :**

| | Qemu, Conteneur | Hôte |
|---|---|---|
| Qemu, Conteneur | Oui | Oui si mise en place d'une ip sur le bridge + route par défaut |

**Configuration pontée (public bridge)**

**Question 7**

**sudo vim /etc/netplan/50-cloud-init.yaml :**

```
network:
  ethernets:
    enp0s3:
      dhcp4: no
    enp0s8:
      dhcp4: true
  bridges:
    br0:
```

```
      addresses: [ 192.168.50.1/24 ]
    br1:
      dhcp4: yes
      interfaces:
            - enp0s3
```

sudo netplan apply

**Adresse ip du bridge : 10.0.2.16/24**

**Question 8**

sudo qemu-system-x86_64 -hda alpine_tp4_1.qcow2 -netdev bridge,id=net0,br=br1 -device e1000,netdev=net0,mac=52:54:00:12:34:55 -k fr -m 128M -vnc :0 &

**Alpine 1 :**
Mode : DHCP
IP : 10.0.2.17
Gateway : 10.0.2.2

**Alpine 2 :**
Mode : DHCP
IP : 10.0.2.18
Gateway : 10.0.2.2

**Question 10**

sed -i 's/br0/br1/g' /var/lib/lxc/debian_1/config /var/lib/lxc/debian_2/config
sed -i 's/lxc.net.0.ipv4*/#&/g' /var/lib/lxc/debian_1/config /var/lib/lxc/debian_2/config

**Debian 1 :**
Mode : DHCP
IP : 10.0.2.19
Gateway : 10.0.2.2

**Debian 2 :**
Mode : DHCP
IP : 10.0.2.20
Gateway : 10.0.2.2

**Question 9,11,12**

**Communications :**

| | Qemu, Conteneur | Hôte | Internet |
|---|---|---|---|
| Qemu, Conteneur | Oui | Oui | Oui |

**Configuration NAT**

**Question 13**

**Donner une adresse ip au bridge :**

sudo ip addr add 192.168.50.1/24 broadcast 192.168.50.255 dev br0

**Configuration du mode Bridge NAT (si besoin d'avoir accès à internet) :**

sudo iptables -t nat -A POSTROUTING -o br1 -j MASQUERADE : ou br1 est l'interface nat de sortie de mon hôte qemu.

**Question 14**

**Alpine 1 :**
Mode : satic
IP : 192.168.50.2
Gateway : 192.168.50.1

**Alpine 2 :**
Mode : satic
IP : 192.168.50.3
Gateway : 192.168.50.1

**Question 16**

sed -i 's/br1/br0/g' /var/lib/lxc/debian_1/config /var/lib/lxc/debian_2/config
sed -i 's/#\(lxc.net.0.ipv4*\)/\1/g' /var/lib/lxc/debian_1/config /var/lib/lxc/debian_2/config

**Debian 1 :**
Mode : satic
IP : 192.168.50.4

Gateway : 192.168.50.1

**Debian 2 :**
Mode : satic
IP : 192.168.50.5
Gateway : 192.168.50.1

**Question 15,17,18**

**Communications :**

| | Qemu, Conteneur | Hôte | Internet |
|---|---|---|---|
| Qemu, Conteneur | Oui | Oui | Oui |

## Bridges en ligne de commande

**sudo vim /etc/netplan/50-cloud-init.yaml :**

```
network:
   ethernets:
      enp0s3:
         dhcp4: no
      enp0s8:
         dhcp4: true
```

sudo netplan apply

**Configuration réseau privé hôte (private bridge)**

```
sudo ip link add name br0 type bridge
sudo ip link set dev br0 up
sudo ip addr add 192.168.50.1/24 broadcast 192.168.50.255 dev br0
```

**Configuration pontée (public bridge)**

```
sudo kill -9 $(pgrep -f dhclient)
sudo ip addr flush dev enp0s3
sudo ip link add name br1 type bridge
sudo ip link set up dev br1
```

sudo ip link set master br1 dev enp0s3
sudo dhclient br1

**Configuration NAT**

sudo iptables -t nat -A POSTROUTING -o br1 -j MASQUERADE

## OpenVSwitch

sudo apt install openvswitch-switch openvswitch-common

**Configuration réseau privé hôte (private bridge)**

sudo ovs-vsctl add-br br0
sudo ip link set up dev br0
sudo ip link set up dev ovs-system
sudo ip addr add 192.168.50.1/24 broadcast 192.168.50.255 dev br0

**Configuration pontée (public bridge)**

sudo ovs-vsctl add-br br1
sudo ip link set up dev br1
sudo ip link set master br1 dev enp0s3
sudo ovs-vsctl add-port br1 enp0s3
sudo dhclient br1

**Configuration NAT**

sudo iptables -t nat -A POSTROUTING -o br1 -j MASQUERADE