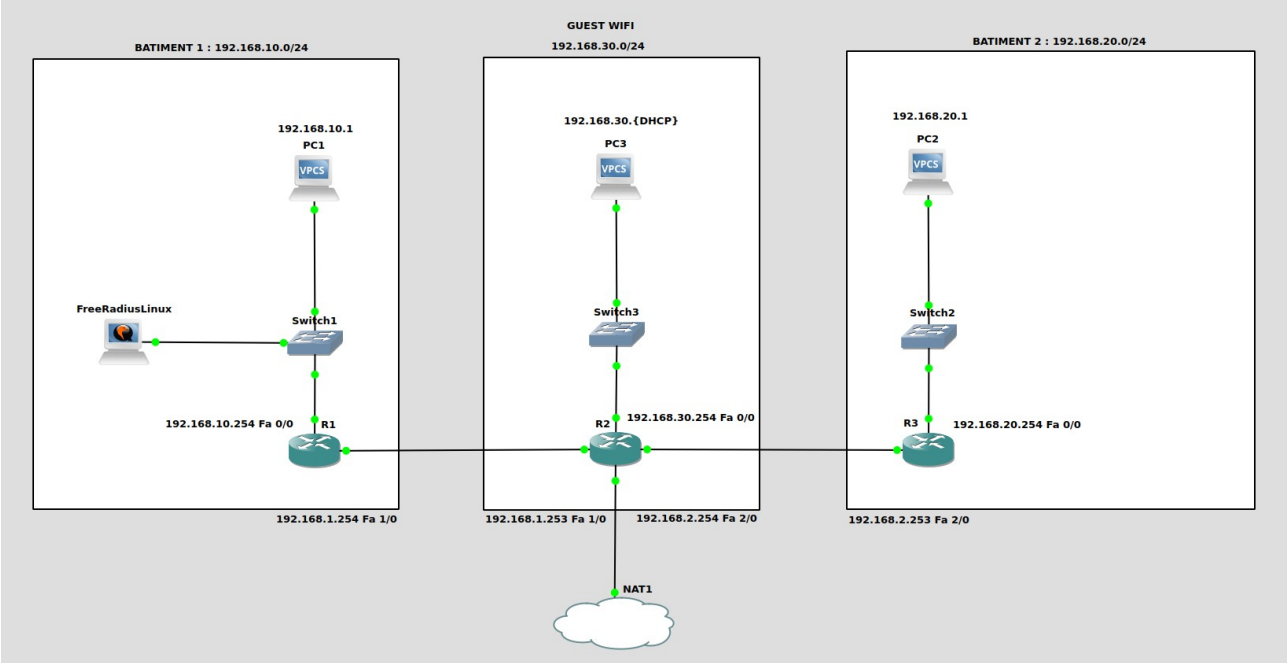
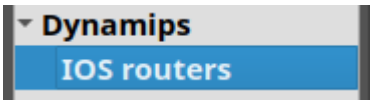
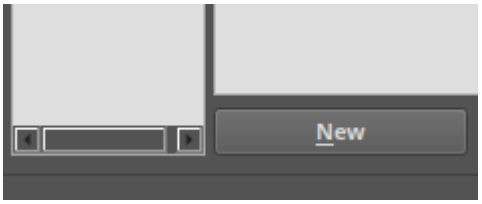
Procédure TP CESI – Infrastructure Cloud



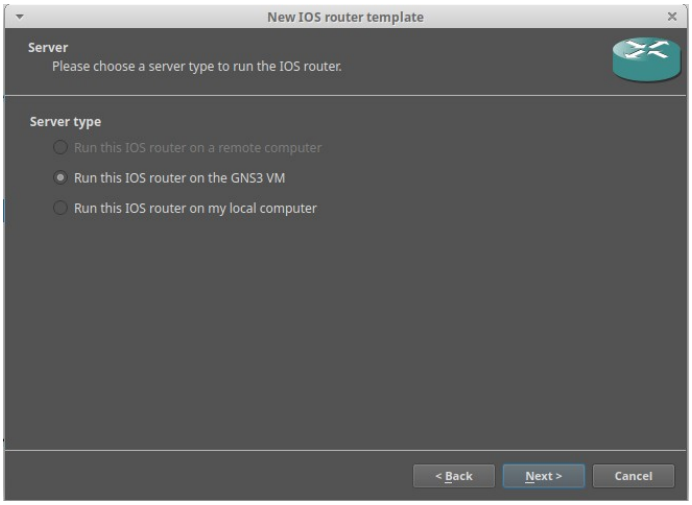
Ouvrir GNS3, aller dans la catégorie ***Edit*** puis ***Preference***.
Positionner vous dans l'onglet ***Dynamips → IOS Routers***.



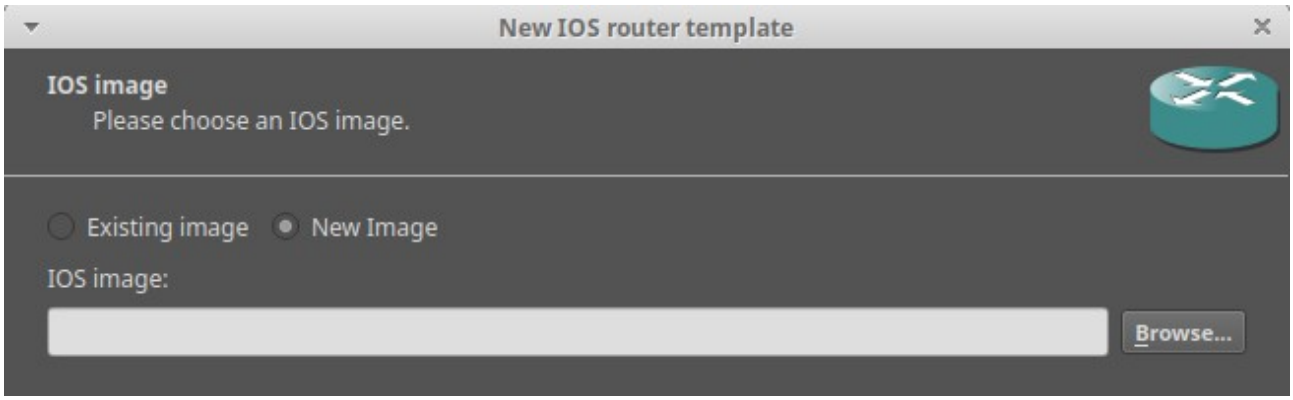
Cliquer sur le bouton ***New*** situer en bas de l'onglet.



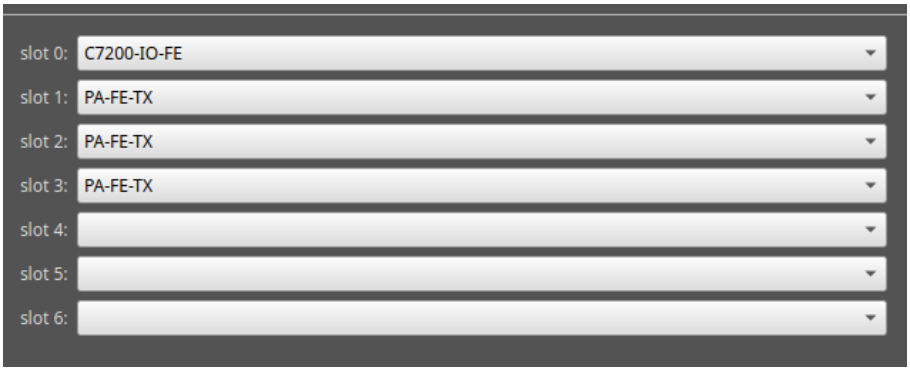
Cocher le choix ***Run this IOS router on the GNS3 VM*** et cliquer sur ***Next***.



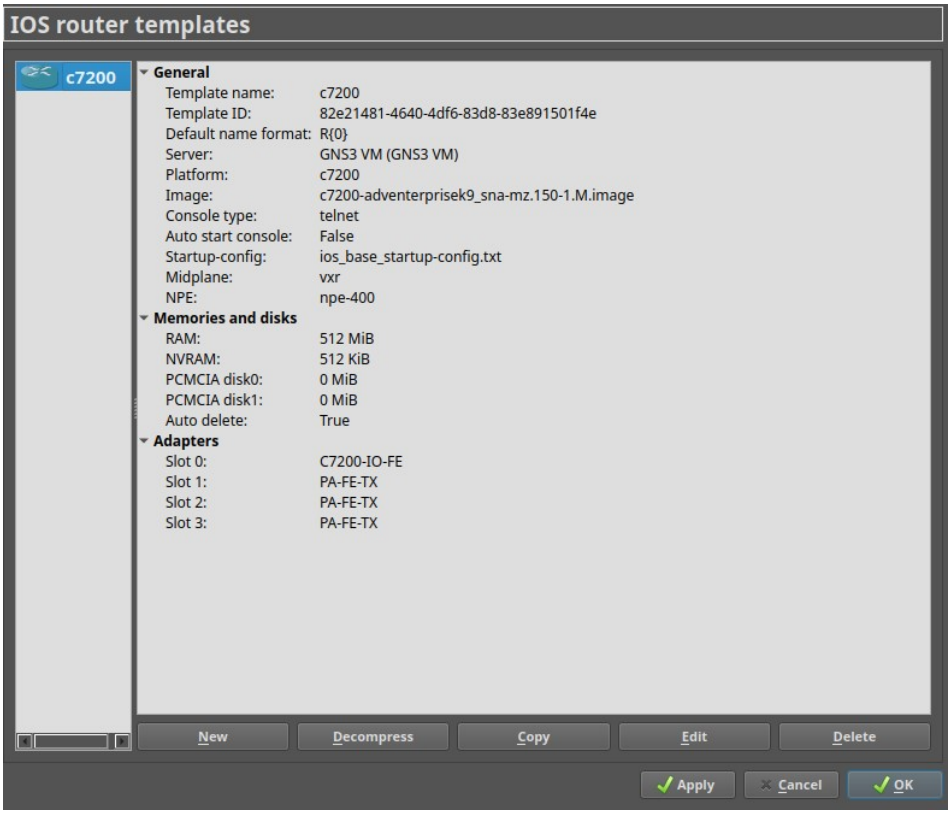
Cocher le choix ***New Image*** et cliquer sur le bouton ***Browse***.
Prendre ***c7200-adventerprisek9_sna-mz.150-1.M.image***.



Cliquer sur le bouton ***Next*** 3 fois (Ignore Name and platform, Memory) et définir slot 1 à 3 en ***PA-FE-TX***.

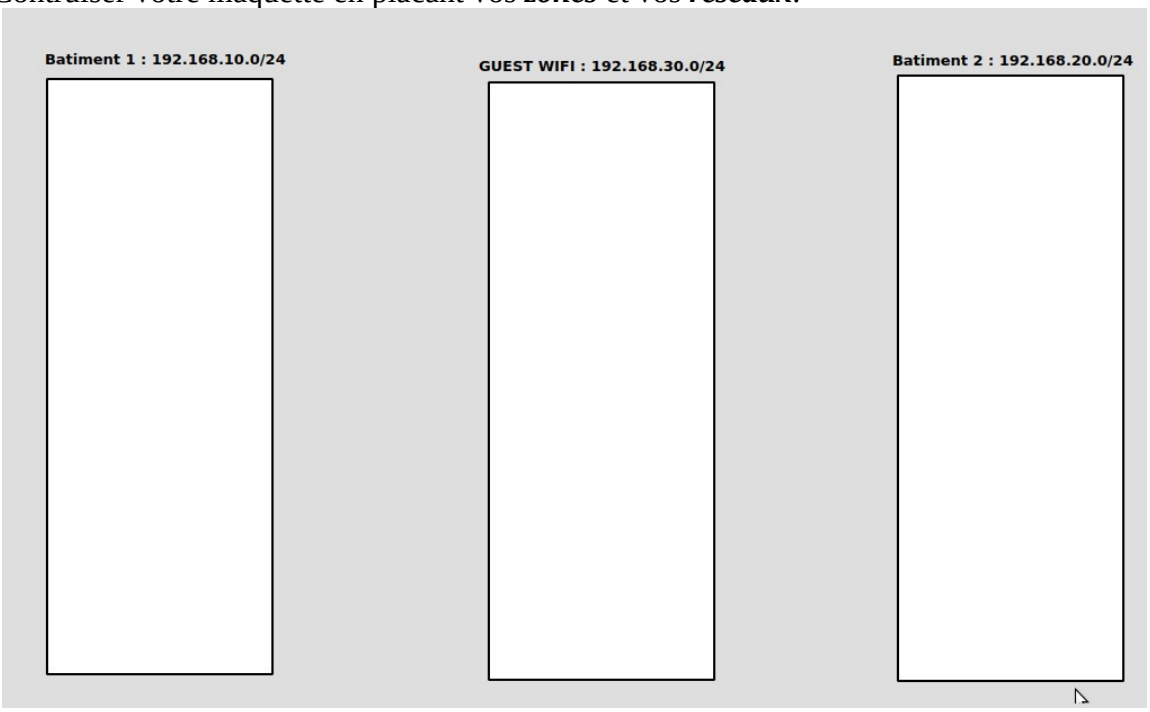


Cliquer sur le bouton ***Next → Finish → Apply → OK***.

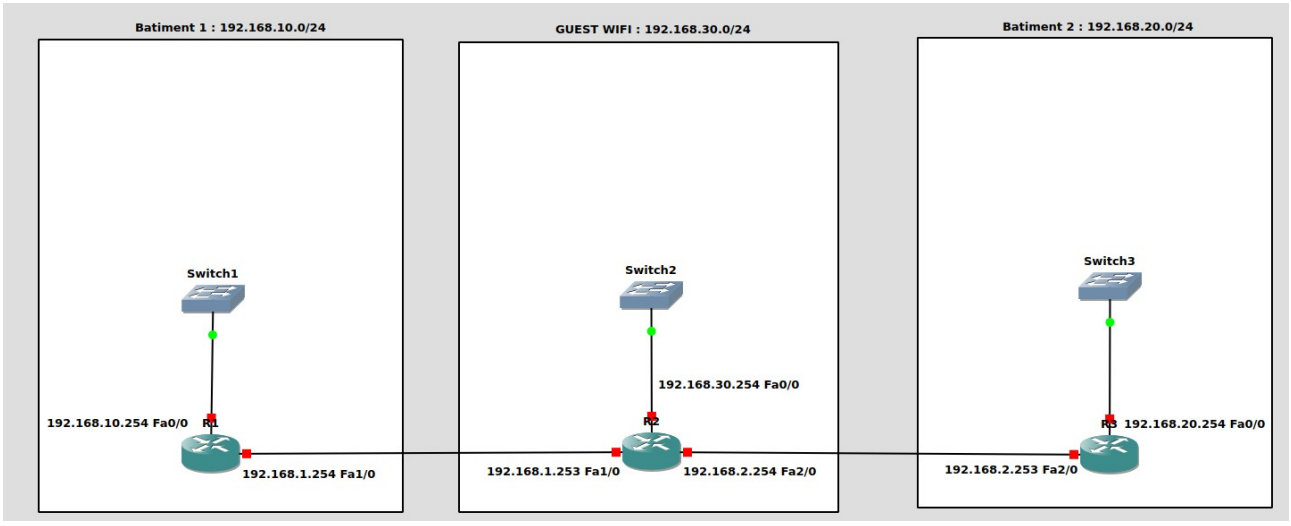


Dans la barre latéral gauche cliquer sur le bouton comportant 4 icones pour afficher l'ensemble des équipements.

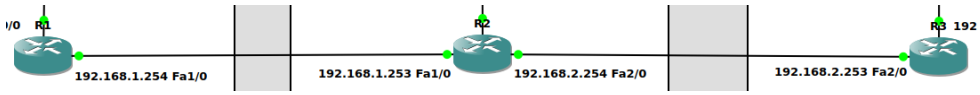
Contruiser votre maquette en placant vos ***zones*** et vos ***réseaux***.



Maintenant placer vos ***switch*** et ***routeurs*** en affichant leurs ***@IP*** et leurs futur ***interface*** pour mieux se répérer sur la ***maquette***.



Démarrer chaque ***routeur*** en faisant un ***clique droit*** dessus puis ***cliquer*** sur le bouton ***Start***.
Les ***leds rouge*** deviendront ***verte*** lorsque vos ***routeur*** seront ***allumer***.



Configurons le routeur ***R1***.
Taper les commandes suivantes dans le routeur ***R1*** via sa ***console*** :

[Voir Annexe ***R1***]

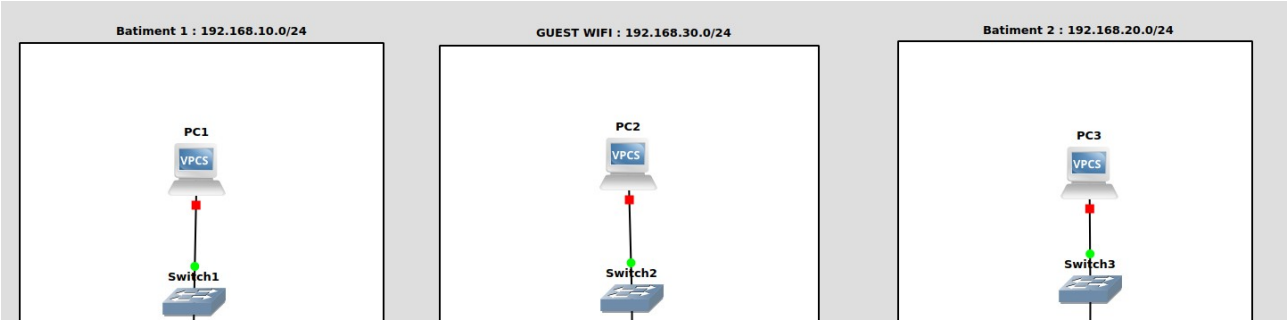
Configurons le routeur ***R2***.
Taper les commandes suivantes dans le routeur ***R2*** via sa ***console*** :

[Voir Annexe ***R2***]

Configurons le routeur ***R3***.
Taper les commandes suivantes dans le routeur ***R3*** via sa ***console*** :

[Voir Annexe ***R3***]

Après configurer nos ***routeur***, vérifions si nos ***réseaux*** peuvent communiquer.
Pour cela ajouter ***3 VPCS*** sur votre ***maquette***.



Démarrer les ***3 VPCS*** et ajouter leur une ***adresse IP*** et la ***Passerelle*** avec la commande suivante :
Note : les @IPs se réinitialise quans vos VPCS sont éteint.

Pcx > ip @IP/@MASQ GATEWAY
PC1 > ip 192.168.10.1/24 192.168.10.254
PC2 > ip dhcp
PC3 > ip 192.168.20.1/24 192.168.20.254

Pour chaque ***VPCS***, essayer de ***ping*** la passerelle et vos voisins.

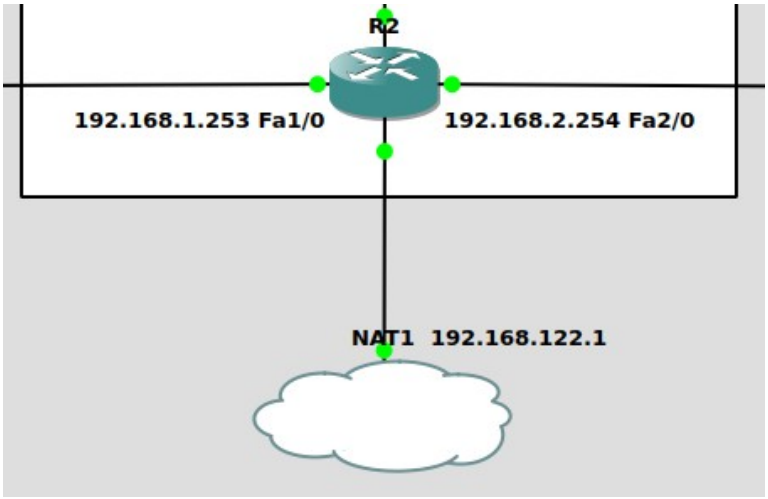
```
PC1> ping 192.168.10.254

84 bytes from 192.168.10.254 icmp_seq=1 ttl=255 time=1.999 ms
84 bytes from 192.168.10.254 icmp_seq=2 ttl=255 time=8.807 ms
84 bytes from 192.168.10.254 icmp_seq=3 ttl=255 time=5.227 ms
84 bytes from 192.168.10.254 icmp_seq=4 ttl=255 time=8.174 ms
```

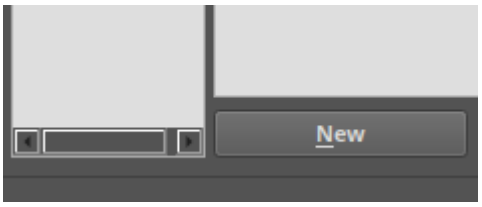
```
PC1> ping 192.168.20.1

84 bytes from 192.168.20.1 icmp_seq=1 ttl=62 time=41.606 ms
84 bytes from 192.168.20.1 icmp_seq=2 ttl=62 time=43.006 ms
84 bytes from 192.168.20.1 icmp_seq=3 ttl=62 time=48.234 ms
84 bytes from 192.168.20.1 icmp_seq=4 ttl=62 time=54.044 ms
```

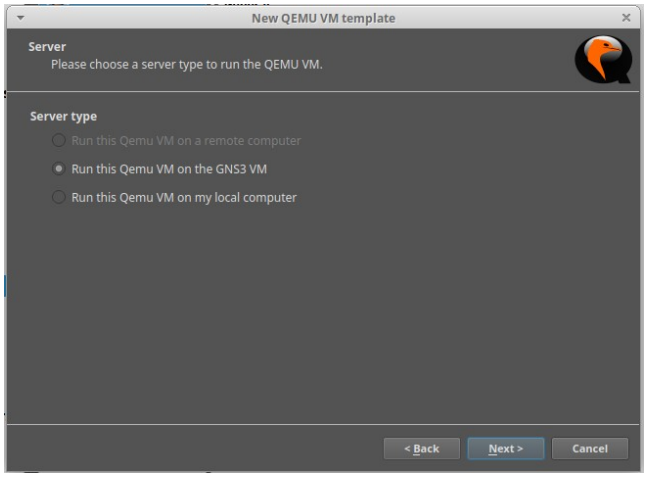
Maintenant que nos réseaux sont ***interconnecté*** et configurer avec une ***liaison VPN*** entre le réseau ***192.168.10.0/24*** et ***192.168.20.0/24***, configurons ***l'accès internet*** pour notre futur serveur linux ***Freeradius***.
Ajouter l'icone nuage nommé ***NAT*** et connecter le avec le routeur ***R2*** sur l'interface ***Fa3/0***.



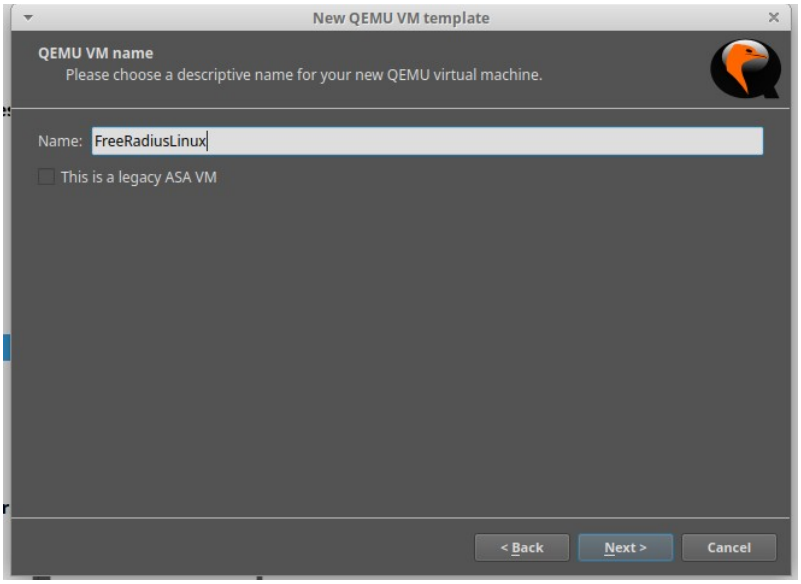
Après avoir mis en place l'accès ***Internet*** pour nos réseaux, installer votre machine virtuelle Linux dans votre topologie GNS3.
Vous pouvez utiliser ***VMWare***, ***VirtualBox*** ou ***QEMU***.
Dans le cadre de ce document, l'option ***QEMU*** sera utiliser sinon *Google est votre amis*.
Retourner dans ***Edit → Preference*** et positionner vous dans l'onglet ***QEMU VMs***.
Cliquer sur le bouton ***New*** situer en bas de l'onglet.



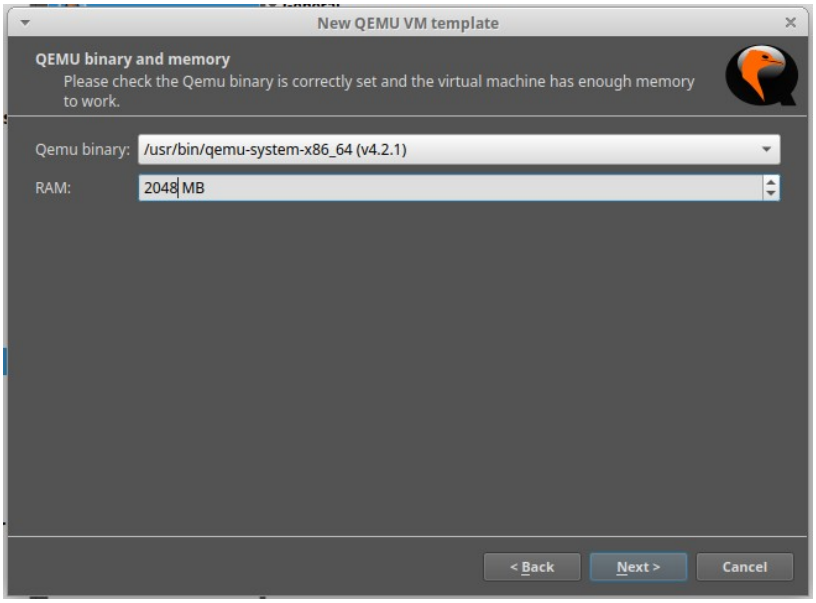
Cocher le choix ***Run this Qemu VM on the GNS3 VM*** et cliquer sur ***Next***.



Nommer votre Machine ***FreeRadiusLinux*** et cliquer sur le bouton ***Next***.

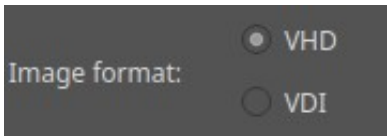


Définissez la ***RAM*** a ***2048 MB*** et cliquer sur le bouton ***Next***.
Votre VM ***GNS3 VM*** doit avoir au ***minimum 2048 MB*** de RAM et ***2 vCPU*** pour pouvoir ***supporter*** l'ajout du serveur Linux
Configuration de GNS3 recommander : ***4 vCPU*** et ***4096 MB*** de RAM.

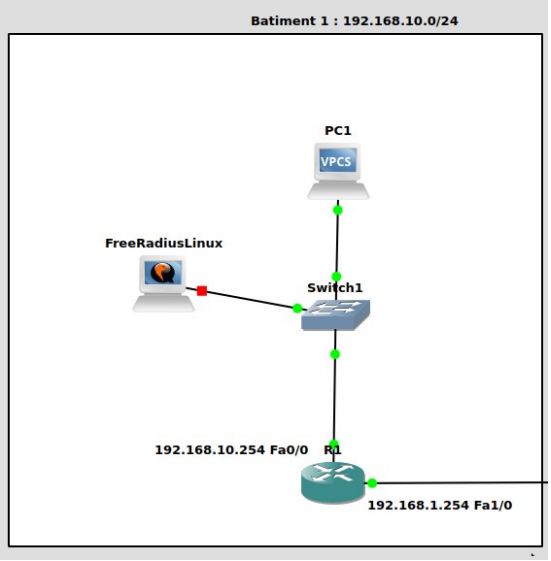


Cliquer sur le bouton ***Next*** et cocher la case ***New Image***, puis le bouton ***Create***.

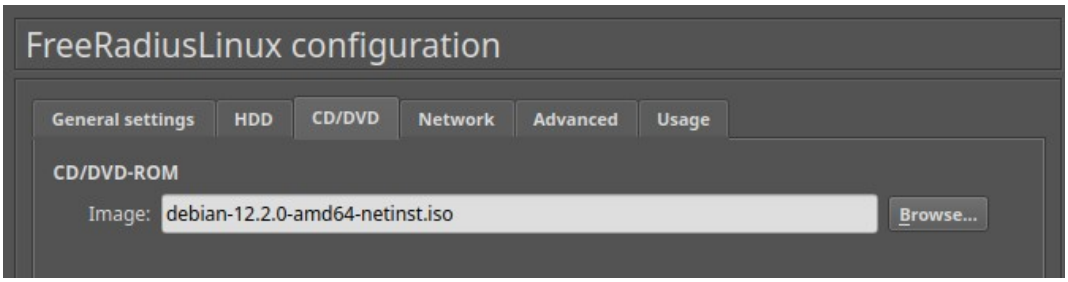
Cocher le choix ***VHD*** dans ***Image Format :*** et cliquer sur le bouton ***Next***.



Cocher le choix ***Dynamic*** et cliquer sur bouton ***Next*** et ***Finish → Finish → Apply → OK***.
Placer votre serveur ***FreeradiusLinux*** sur votre ***maquette*** et connecter le au ***switch***.



Faite un ***clique droit*** sur votre ***serveur*** et cliquer sur ***Configure***.
Définissez le type de console par ***VNC***.
Aller dans l'onglet ***CD/DVD*** et cliquer sur ***Browse*** et choissir un ***ISO Linux → Apply → OK***.



Démarrer votre serveur ***FreeradiusLinux*** et faite une installation classique Linux.

Configurer ***l'interface réseau*** de votre serveur ***FreeradiusLinux*** et redémarrer le ***service réseau***.

```
# The primary network interface
allow-hotplug ens3
iface ens3 inet static
address 192.168.10.10
netmask 255.255.255.0
gateway 192.168.10.254
```

/etc/init.d/networking restart

Sur votre serveur ***FreeradiusLinux***, essayer de ***ping*** la passerelle et vos voisins.

```
root@debian:~# ping 192.168.10.254
PING 192.168.10.254 (192.168.10.254) 56(84) bytes of data.
64 bytes from 192.168.10.254: icmp_seq=1 ttl=255 time=9.16 ms
64 bytes from 192.168.10.254: icmp_seq=2 ttl=255 time=10.3 ms
```

```
root@debian:~# ping 192.168.20.1
PING 192.168.20.1 (192.168.20.1) 56(84) bytes of data.
64 bytes from 192.168.20.1: icmp_seq=1 ttl=62 time=37.1 ms
64 bytes from 192.168.20.1: icmp_seq=2 ttl=62 time=36.4 ms
```

Après que notre serveur ***FreeradiusLinux*** soit bien connecter dans le réseau, installer le paquet ***freeradius*** via ***APT***.

```
root@debian:~# apt install freeradius_
```

Ajoutons nos ***réseaux*** comme client ***RADIUS***.
Aller dans le fichier ***/etc/freeradius/3.0/clients.conf*** et ajouter les lignes suivante :

```
client 192.168.10.0 {
ipaddr = 192.168.10.0/24
secret = cesi123
}

client 192.168.20.0 {
ipaddr = 192.168.20.0/24
secret =cesi123
}

client 192.168.30.0 {
ipaddr = 192.168.30.0/24
secret = cesi123
}

client 192.168.1.0 {
ipaddr = 192.168.1.0/24
secret = cesi123
}

client 192.168.2.0 {
ipaddr = 192.168.2.0/24
secret = cesi123
}
```

Maintenant ajoutons nos ***utilisateurs***.
Aller dans le fichier ***/etc/freeradius/3.0/users*** et ajouter les lignes suivante :

```
antonin Cleartext-Password := "cesi123"
        Reply-Message = "Bienvenue %{User-Name} sur le routeur !"

nathan Cleartext-Password := "boss123"
        Reply-Message = "Bienvenue %{User-Name} sur le routeur !"
```

Après avoir ***configurer*** nos clients et utilisateurs, ***redémarrer*** le service ***Freeradius***.

```
root@debian:~# service freeradius restart
root@debian:~#
```

Ensuite essayer de vous connecter en ***SSH*** sur ***R1*** avec un utilisateur ***RADIUS*** avec la commande suivante :

ssh -oKexAlgorithms=+diffie-hellman-group1-sha1 -oHostkeyAlgorithms=+ssh-rsa -l <user-radius> <@IP Cisco>

```
root@debian:~# ssh -oKexAlgorithms=+diffie-hellman-group1-sha1 -oHostkeyAlgorithms=+ssh-rsa -l antonin 192.168.10.254
(antonin@192.168.10.254) Password:
Bienvenue antonin sur le routeur !

R1>_
```

Précision : Lorsqu'on active l'authentification ***RADIUS*** sur un routeur, cela ***désactive*** l'authentification ***LOCAL*** dont le compte ***admin*** !

Pour voir le trafic ***VPN*** entre le réseau ***192.168.10.0/24*** et ***192.168.20.0/24*** :

show crypto engine connection active

```
R1#show crypto engine connection active
Crypto Engine Connections

   ID  Type    Algorithm           Encrypt  Decrypt LastSeqN IP-Address
    1  IPsec   AES+SHA                   0        5        5 192.168.1.254
    2  IPsec   AES+SHA                   5        0        0 192.168.1.254
 1001  IKE     SHA+AES                   0        0        0 192.168.1.254
```

Annexe 1 (Conf Cisco R1) :

```
enable
configure terminal
hostname R1
ip domain-name cesi
crypto key generate rsa general-keys modulus 2048
ip ssh version 2
service password-encryption
username admin privilege 15 password 0 cesi
line vty 0 4
login local
transport input telnet sshk
exit
interface FastEthernet 0/0
no shutdown
ip address 192.168.10.254 255.255.255.0
ip nat inside
exit
interface FastEthernet 1/0
no shutdown
ip address 192.168.1.254 255.255.255.0
ip nat outside
exit
ip route 0.0.0.0 0.0.0.0 192.168.1.253
router ospf 1
network 192.168.10.0 0.0.0.255 area 0
network 192.168.1.0 0.0.0.255 area 0
exit
access-list 100 deny ip 192.168.10.0 0.0.0.255 192.168.20.0 0.0.0.255
access-list 100 permit ip 192.168.10.0 0.0.0.255 any
ip nat inside source list 100 interface FastEthernet 1/0 overload
crypto isakmp enable
crypto isakmp policy 10
encryption aes
authentication pre-share
hash sha
group 2
lifetime 86400
exit
crypto isakmp key cesi address 192.168.2.253
crypto ipsec transform-set VPNLABO esp-aes esp-sha-hmac
crypto ipsec security-association lifetime seconds 86400
ip access-list extended VPN
permit ip 192.168.10.0 0.0.0.255 192.168.20.0 0.0.0.255
exit
crypto map CARTEVPN 10 ipsec-isakmp
match address VPN
set peer 192.168.2.253
set transform-set VPNLABO
exit
interface FastEthernet 1/0
crypto map CARTEVPN
exit
aaa new-model
aaa authentication dot1x default group radius
aaa authentication login default group radius
aaa authorization network default group radius
aaa authorization exec default group radius
radius-server host 192.168.10.10 auth-port 1812 acct-port 1646 key cesi123
do wr
```

Annexe 2 (Conf Cisco R2) :

```
enable
configure terminal
hostname R2
ip domain-name cesi
crypto key generate rsa general-keys modulus 2048
ip ssh version 2
service password-encryption
username admin privilege 15 password 0 cesi
line vty 0 4
login local
transport input telnet ssh
exit
interface FastEthernet 0/0
no shutdown
ip address 192.168.30.254 255.255.255.0
ip nat inside
exit
interface FastEthernet 1/0
no shutdown
ip address 192.168.1.253 255.255.255.0
ip nat inside
exit
interface FastEthernet 2/0
no shutdown
ip address 192.168.2.254 255.255.255.0
ip nat inside
exit
interface FastEthernet 3/0
no shutdown
ip address dhcp
ip nat outside
exit
access-list 100 permit any
ip nat inside source list 100 interface fastEthernet 3/0 overload
ip route 0.0.0.0 0.0.0.0 192.168.122.1
ip dhcp pool GUEST-WIFI
network 192.168.30.0 255.255.255.0
domain-name cesi
dns-server 8.8.8.8
lease 0 8
exit
ip dhcp excluded-address 192.168.30.1 192.168.30.10
router ospf 1
network 192.168.1.0 0.0.0.255 area 0
network 192.168.2.0 0.0.0.255 area 0
network 192.168.30.0 0.0.0.255 area 0
do wr
```

Annexe 3 (Conf Cisco R3) :

```
enable
configure terminal
hostname R3
ip domain-name cesi
crypto key generate rsa general-keys modulus 2048
ip ssh version 2
service password-encryption
username admin privilege 15 password 0 cesi
line vty 0 4
login local
transport input telnet ssh
exit
interface FastEthernet 0/0
no shutdown
ip address 192.168.20.254 255.255.255.0
ip nat inside
exit
interface FastEthernet 2/0
no shutdown
ip address 192.168.2.253 255.255.255.0
ip nat outside
exit
ip route 0.0.0.0 0.0.0.0 192.168.2.254
router ospf 1
network 192.168.20.0 0.0.0.255 area 0
network 192.168.2.0 0.0.0.255 area 0
exit
access-list 100 deny ip 192.168.20.0 0.0.0.255 192.168.10.0 0.0.0.255
access-list 100 permit ip 192.168.20.0 0.0.0.255 any
ip nat inside source list 100 interface FastEthernet 2/0 overload
crypto isakmp enable
crypto isakmp policy 10
encryption aes
authentication pre-share
hash sha
group 2
lifetime 86400
exit
crypto isakmp key cesi address 192.168.1.254
crypto ipsec transform-set VPNLABO esp-aes esp-sha-hmac
crypto ipsec security-association lifetime seconds 86400
ip access-list extended VPN
permit ip 192.168.20.0 0.0.0.255 192.168.10.0 0.0.0.255
exit
crypto map CARTEVPN 10 ipsec-isakmp
match address VPN
set peer 192.168.1.254
set transform-set VPNLABO
exit
interface FastEthernet 2/0
crypto map CARTEVPN
exit
do wr
```