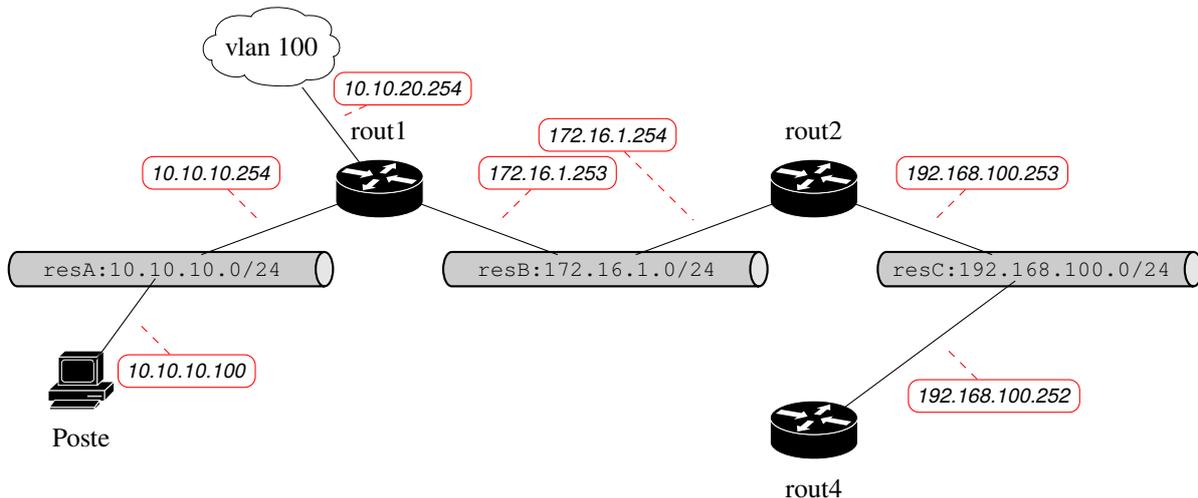


Routage dynamique avec RIP

■ ■ ■ Routage à l'aide de RIP



**But de la simulation**

- ▷ configurer et déployer le protocole RIP ;
- ▷ analyser les paquets échangés et le fonctionnement du protocole ;

**Mise en place de la simulation :**

- \* On utilisera le logiciel frr qui peut mettre en œuvre les protocoles BGP, OSPF, OSPF6 (en IPv6), RIP, RIPng (en IPv6) et IS-IS ;
- \* les routeurs rout1, rout2 et rout3 vont être simulés par des « netns » ;
- \* la machine « Poste » va être simulée par un netns ;
- \* les 3 réseaux resA, resB et resC vont être simulés par des switches (open-vSwitches sous Linux) ;
- \* le VLAN correspondra uniquement à une interface du routeur rout1 sans connexion extérieure réelle ;
- \* le protocole de routage RIP va être mis en œuvre par frr (cf cours d'Infrastructure Réseaux).

**Pour l'installation d'Open-vSwitch :**

```
xterm
rez@ishtar:~$ sudo apt install openvswitch-switch
```

**Pour l'installation de frr sur la VM :**

```
xterm
rez@ishtar:~$ sudo apt install frr
```

On va configurer l'outil frr de manière à ce qu'il **ne tourne pas** sur le « netns root », c-à-d la pile TCP/IP de la machine (il ne tournera que dans les netns de simulation) :

```
xterm
rez@ishtar:~$ sudo systemctl stop frr
rez@ishtar:~$ sudo systemctl disable frr
```

Le service frr ne démarrera plus automatiquement au démarrage de la machine.

On aura également besoin de l'outil « telnet » pour la configuration de frr :

```
xterm
rez@ishtar:~$ sudo apt install telnet
```

On va configurer frr pour la gestion des netns, en éditant le fichier `/etc/frr/daemons`

```
xterm
...
# The list of daemons to watch is automatically generated by the init script.
#watchfrr_options=""

# To make watchfrr create/join the specified netns, use the following option:
watchfrr_options="--netns".----- la ligne à décommenter
# This only has an effect in /etc/frr/<somename>/daemons, and you need to
# start FRR with "/usr/lib/frr/frrinit.sh start <somename>".
...
```

### Récupération du fichier de création des différents netns, switches et poste

Vous pourrez récupérer le fichier de configuration :

```
xterm
rezo@ishtar:~$ git clone https://git.p-fb.net/PeFClic/infra_lab
```

▷ le script «`build_architecture`» permet de créer l'ensemble des netns, des liens, des switches et assure la configuration des interfaces ;

```
xterm
rezo@ishtar:~/INFRA_LAB$ sudo ./build_architecture
```

▷ le script «`clean`» permet de supprimer l'ensemble des éléments créés précédemment :

```
xterm
rezo@ishtar:~/INFRA_LAB$ sudo ./clean
```

*Cela vous permet de recréer entièrement le «netlab».*

**Rappel :** pour accéder à un netns, vous pouvez exécuter un shell dedans :

```
xterm
rezo@ishtar:~/INFRA_LAB$ sudo ip netns exec poste bash
```

*Ici, j'accède au netns «poste» Ou plus simplement :*

```
xterm
rezo@ishtar:~/INFRA_LAB$ ./netns poste
rezo@ishtar:~/INFRA_LAB$ [poste]
```

### Configuration des trois netns, appelés «`rout1`», «`rout2`» et «`rout4`» :

On exécute avec les droits «`root`», le script `init_frr_rip` qui va s'occuper de configurer les netns :

```
xterm
rezo@ishtar:~$ sudo ./init_frr_rip
```

## Que fait le script `init_frr_rip` ?

Chacun de ces netns va exécuter `frr` : il faut éviter des **collisions** au niveau des fichiers de configuration et des fichiers temporaires, c-à-d que chaque netns disposera de répertoires particuliers qui vont « remplacer » les répertoires et fichiers présents dans le système de fichiers :

- ▷ pour un netns appelé `rout1`, le répertoire « `/etc/netns/rout1/frr` » remplacera le répertoire « `/etc/frr` » pour le netns.
- ▷ pour chaque netns, un répertoire « `/etc/frr` » va être créé : par exemple pour `rout1`, on va le créer dans « `/etc/netns/rout1` », ce qui donnera le répertoire « `/etc/netns/rout1/frr` ».
- ▷ pour le bon fonctionnement de `frr`, il faut également créer un sous-répertoire au nom du netns dans le répertoire précédent..
- ▷ pour faire fonctionner le démon s'occupant de `rip`, il faut une copie dans le répertoire associé au netns des deux fichiers suivants :

le fichier « `daemons` »

```
zebra=yes ----- yes
bgpd=no
ospfd=no
ospf6d=no
ripd=yes ----- yes
ripngd=no
isisd=no
pimd=no
ldpd=no
...
```

Le fichier « `ripd.conf` » :

```
hostname Router
password zebra
```

*C'est nécessaire pour permettre la connexion par telnet vers l'interface de configuration du démon.*

- ▷ On donne les droits d'accès à ces fichiers à l'utilisateur « `frr` » pour lui permettre de sauver la configuration du démon une fois qu'elle a été faite.
- ▷ Le script qui s'occupe de tout ça est dans le git, il porte le nom de `init_frr_rip` On va créer le répertoire « `/etc/netns/routi/frr/routi` » :

```
xterm
#!/bin/bash -x

for i in $(seq 1 4)
do
  mkdir -p /etc/netns/rout${i}/frr/rout${i}
  cp frr/* /etc/netns/rout${i}/frr/rout${i}/
  chown -R frr:frr /etc/netns/rout${i}/frr
done
```

créer le répertoire  
copier les fichiers de config  
donner les droits

- ▷ On vérifie que les différents fichiers existent bien pour chaque netns avec les bons droits :

```
xterm
rezo@ishtar:~/INFRA_LAB$ sudo ls -lR /etc/netns/rout1
with pef at ishtar at 12:55:03
/etc/netns/rout1:
total 4
drwxr-xr-x 3 frr frr 4096 Mar 28 23:19 frr

/etc/netns/rout1/frr:
total 4
drwxr-xr-x 2 frr frr 4096 Mar 30 23:05 rout1

/etc/netns/rout1/frr/rout1:
total 16
-rw-r--r-- 1 frr frr 1514 Mar 31 12:54 daemons
-rw-r--r-- 1 frr frr 31 Mar 31 12:54 ripd.conf
```

## ■ ■ ■ Configuration de frr

On va :

- se mettre dans le netns ;
- démarrer le démon ripd avec le script `/usr/lib/frr/frrinit.sh` :
  - ▷ `sudo /usr/lib/frr/frrinit.sh start rout1` pour démarrer les démons ;
  - ▷ `sudo /usr/lib/frr/frrinit.sh stop rout1` pour stopper les démons ;

```
xterm
rezo@ishtar:~/INFRA_LAB$ ./netns rout1
rezo@ishtar:~/INFRA_LAB$ [rout1] sudo ss -tlnp
State          Recv-Q          Send-Q          Local Address:Port
Peer Address:Port Process
pef@ishtar:~/INFRA_LAB$ [rout1] sudo /usr/lib/frr/frrinit.sh start rout1
* Started watchfrr
pef@ishtar:~/INFRA_LAB$ [rout1] sudo ss -tlnp
State          Recv-Q          Send-Q          Local Address:Port  Peer Address:Port
Process
LISTEN        0                3                127.0.0.1:2601      0.0.0.0:*
users: (("zebra", pid=28759, fd=27))
LISTEN        0                3                127.0.0.1:2602      0.0.0.0:*
users: (("ripd", pid=28772, fd=12))
LISTEN        0                3                127.0.0.1:2616      0.0.0.0:*
users: (("staticd", pid=28778, fd=12))
pef@ishtar:~/INFRA_LAB$ [rout1]
```

le script netns est dans le dépôt git

le script de démarrage des démons dans le netns

(les démons `zebra` et `staticd` sont nécessaires pour la configuration de la table de routage).

## Configuration de RIP

La configuration du routeur pour le protocole RIP se fait par l'intermédiaire d'une connexion sur le port 2602 associé au démon « RIPd » :

```
xterm
rezo@ishtar:~/INFRA_LAB$ [rout1] telnet
127.0.0.1 2602
Trying 127.0.0.1...
Connected to 127.0.0.1.
Escape character is '^]'.

Hello, this is FRRouting (version 8.1).
Copyright 1996-2005 Kunihiro Ishiguro, et al.

User Access Verification

Password:
Router> enable
Router#
```

le mot de passe est zebra

Une fois connecté, il faut activer le mode « privilégié » avec la commande `enable`.

À tout moment, l'utilisation de la touche « ? » permet l'affichage de l'aide.

La configuration du routeur est similaire à celle utilisée par des matériels fournis par la société CISCO tournant sous le système d'exploitation IOS.

Pour configurer RIP dans tout le réseau, il va falloir :

- a. activer le service `frr` sur chaque netns concerné (`rout1`, `rout2` et `rout4`) avec le script `/usr/lib/frr/frrinit.sh` ;
- b. se connecter au démon `ripd` local avec `telnet` sur le port d'écoute 2602 ;
- c. configurer la liste des réseaux gérés par le routeur dans l'interface de configuration, par l'intermédiaire des commandes suivantes :

<code>network 10.0.0.0/8</code>	sert à demander la gestion par le démon du réseau indiqué
<code>no network 10.0.0.0/8</code>	sert à supprimer la gestion
<code>router rip</code>	passse à la configuration de RIP
<code>default-information originate</code>	permet d'indiquer que le routeur effectuant cette commande devient la route par défaut du réseau (ici, ce sera <code>rout2</code> )
<code>write file</code>	permet de sauvegarder la configuration actuelle du routeur
<code>show running-config</code>	permet d'afficher la configuration courante
<code>show ip rip</code>	permet d'afficher la configuration du protocole RIP du routeur.

## Pour la configuration de rout1 :

```
xterm
rezo@ishtar:~$ sudo ip netns exec rout1 bash
rezo@ishtar:~# telnet 127.0.0.1 2602
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
Hello, this is Quagga (version 0.99.18).
Copyright 1996-2005 Kunihiro Ishiguro, et al.

User Access Verification
Password:
ripd> enable
ripd# configure terminal
ripd(config)# router rip
ripd(config-router)# network 10.10.10.0/24
ripd(config-router)# network 10.10.20.0/24
ripd(config-router)# network 172.16.1.0/24
ripd(config-router)# show running-config

Current configuration:
!
hostname ripd
password zebra
log stdout
!
router rip
 network 10.10.10.0/24
 network 10.10.20.0/24
 network 172.16.1.0/24
!
line vty
!
end
ripd(config-router)# write file
Configuration saved to /etc/quagga/ripd.conf
ripd(config-router)# exit
ripd(config)# exit
ripd# show ip rip
Codes: R - RIP, C - connected, S - Static, O - OSPF, B - BGP
Sub-codes:
      (n) - normal, (s) - static, (d) - default, (r) - redistribute,
      (i) - interface

      Network          Next Hop          Metric From      Tag Time
C(i) 10.10.10.0/24    0.0.0.0           1 self           0
C(i) 10.10.20.0/24    0.0.0.0           1 self           0
C(i) 172.16.1.0/24    0.0.0.0           1 self           0
```

## Pour la configuration de rout4 :

```
rez@ishtar:~/INFRA_LAB$ ./netns rout4
rez@ishtar:~/INFRA_LAB$ [rout4] sudo /usr/lib/frr/frrinit.sh start rout4
* Started watchfrr
rez@ishtar:~/INFRA_LAB$ [rout4] ss -tlnp
State      Recv-Q      Send-Q      Local Address:Port
Peer Address:Port      Process
LISTEN     0            3           127.0.0.1:2601
0.0.0.0:*
LISTEN     0            3           127.0.0.1:2602
0.0.0.0:*
LISTEN     0            3           127.0.0.1:2616
0.0.0.0:*
pez@atmos:~/INFRA_LAB$ [rout4] telnet 127.0.0.1 2602
Trying 127.0.0.1...
Connected to 127.0.0.1.
Escape character is '^]'.

Hello, this is FRRouting (version 8.1).
Copyright 1996-2005 Kunihiro Ishiguro, et al.

User Access Verification

Password:
Router> enable
Router# configure terminal
Router(config)# router rip
Router(config-router)# network 192.168.100.0/24
Router(config-router)# write file
Configuration saved to /etc/frr/rout4/ripd.conf
Router(config-router)# show running-config

Current configuration:
!
frr version 8.1
frr defaults traditional
!
hostname Router
password zebra
!
!
!
!
router rip
 network 192.168.100.0/24
exit
!
!
!
!
end
Router(config-router)# exit
Router(config)# exit
Router# show ip rip
Codes: R - RIP, C - connected, S - Static, O - OSPF, B - BGP
Sub-codes:
      (n) - normal, (s) - static, (d) - default, (r) - redistribute,
      (i) - interface

      Network      Next Hop      Metric From      Tag Time
C(i) 192.168.100.0/24  0.0.0.0      1 self          0
Router#
```

## Pour la configuration de rout2 :

```
xterm
rezo@ishtar:~/INFRA_LAB$ ./netns rout2
rezo@ishtar:~/INFRA_LAB$ [rout2] sudo /usr/lib/frr/frrinit.sh start rout2
* Started watchfrr
pef@atmos:~/INFRA_LAB$ [rout2] ss -tlnp
State      Recv-Q      Send-Q
Peer Address:Port      Process
LISTEN    0            3
0.0.0.0:* 127.0.0.1:2616
LISTEN    0            3
0.0.0.0:* 127.0.0.1:2601
LISTEN    0            3
0.0.0.0:* 127.0.0.1:2602
rezo@ishtar:~/INFRA_LAB$ [rout2] telnet 127.0.0.1 2602
Trying 127.0.0.1...
Connected to 127.0.0.1.
Escape character is '^]'.

Hello, this is FRRouting (version 8.1).
Copyright 1996-2005 Kunihiro Ishiguro, et al.

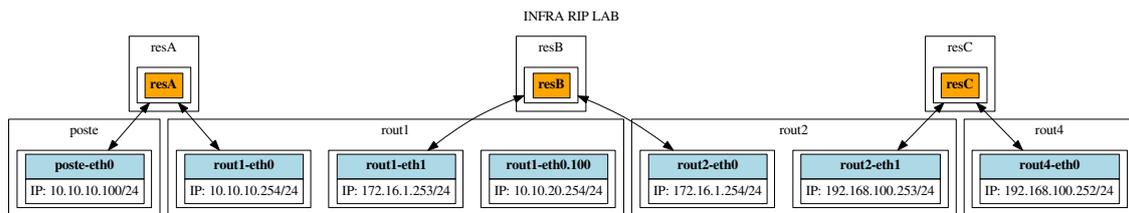
User Access Verification

Password:
Router> enable
Router# configure terminal
Router(config)# router rip
Router(config-router)# network 172.16.1.0/24
Router(config-router)# network 192.168.100.0/24
Router(config-router)# default-information originate
Router(config-router)# show running-config

Current configuration:
!
frr version 8.1
frr defaults traditional
!
hostname Router
password zebra
!
!
!
!
router rip
 default-information originate
  network 172.16.1.0/24
  network 192.168.100.0/24
exit
!
!
!
!
end
Router(config-router)# write file
Configuration saved to /etc/frr/rout2/ripd.conf
Router(config-router)# exit
Router(config)# exit
Router# show ip rip
Codes: R - RIP, C - connected, S - Static, O - OSPF, B - BGP
Sub-codes:
      (n) - normal, (s) - static, (d) - default, (r) - redistribute,
      (i) - interface

      Network          Next Hop          Metric From          Tag Time
R(d) 0.0.0.0/0        0.0.0.0           1 self              0
R(n) 10.10.10.0/24    172.16.1.253      2 172.16.1.253       0 02:57
R(n) 10.10.20.0/24    172.16.1.253      2 172.16.1.253       0 02:57
C(i) 172.16.1.0/24    0.0.0.0           1 self              0
C(i) 192.168.100.0/24 0.0.0.0           1 self              0
Router#
```

*Vous remarquez que dès la configuration de rout2, le protocole RIP s'active et construit la table de routage du réseau complet.*



## ■■■ Travail

1. Sur chaque poste vérifiez les informations de routage construites :

```

xterm
rezo@ishtar:~/INFRA_LAB$ [rout2] ip r
10.10.10.0/24 nhid 10 via 172.16.1.253 dev rout2-eth0 proto rip metric 20
10.10.20.0/24 nhid 10 via 172.16.1.253 dev rout2-eth0 proto rip metric 20
172.16.1.0/24 dev rout2-eth0 proto kernel scope link src 172.16.1.254
192.168.100.0/24 dev rout2-eth1 proto kernel scope link src 192.168.100.253

```

2. à l'aide de tcpdump, *sniffez* et étudiez les paquets du protocole RIP sur les différentes interfaces des différents routeurs : `tcpdump -lnvv udp`.
3. essayez une connexion entre le « rout1 » et « rout2 » avec la commande `socat`.
4. à l'aide de NetFilter donnez l'accès Internet (iptables, table « nat », chaîne POSTROUTING etc.)
  - ◇ pour les réseaux 172.16.1.0/24 et 192.168.100.0/24 ;
  - ◇ pour le réseau 10.10.10.0/24.

Quel est le routeur désigné comme « route par défaut » dans RIP ?

Vous :

- ◇ activez l'interface de l'un des switches connecté à ce routeur sur le « netns root », c-à-d votre VM ou votre machine sous Linux ;
- ◇ configurez une adresse IP correcte pour cette interface ;
- ◇ vous désignerez votre VM/machine Linux comme route par défaut pour le routeur choisi ;
- ◇ vous activez la fonction de routage sur votre VM/machine Linux ;
- ◇ vous configurez le firewall ;
- ◇ vous vérifierez que depuis « poste » vous accédez bien à Internet (ping sur 8.8.8.8).

Comment pouvez vous configurer le DNS sur le netns « poste » ?

*Indice : pouvez vous lui configurer son propre fichier « /etc/resolv.conf » et que devrait-il contenir ?*