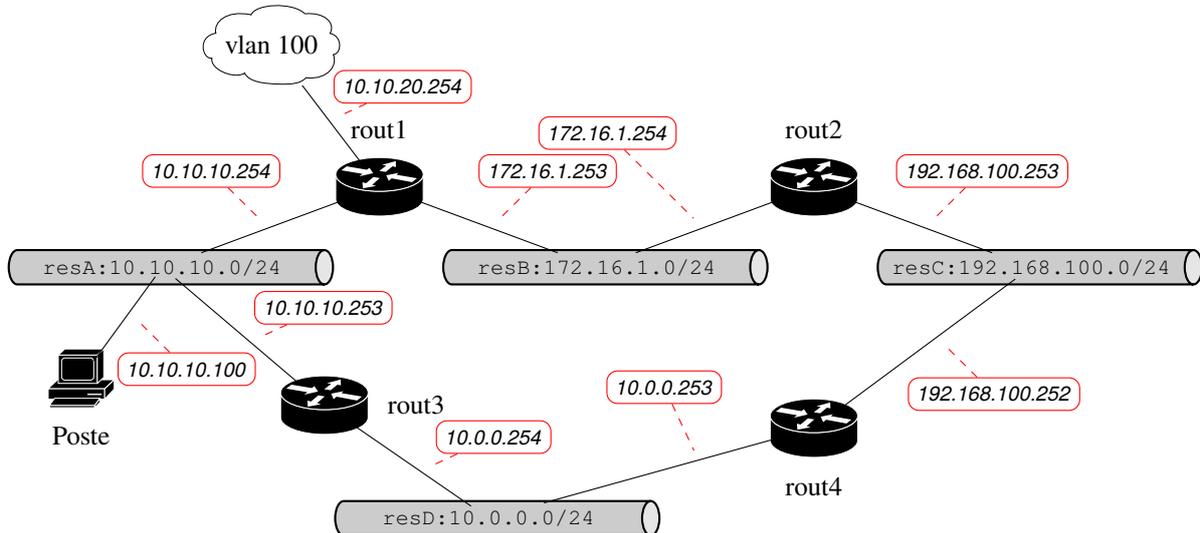


Routage dynamique avec RIP & OSPF

■ ■ ■ Routage à l'aide de RIP & OSPF — Extension du réseau d'interconnexion de la fiche de TP n°1

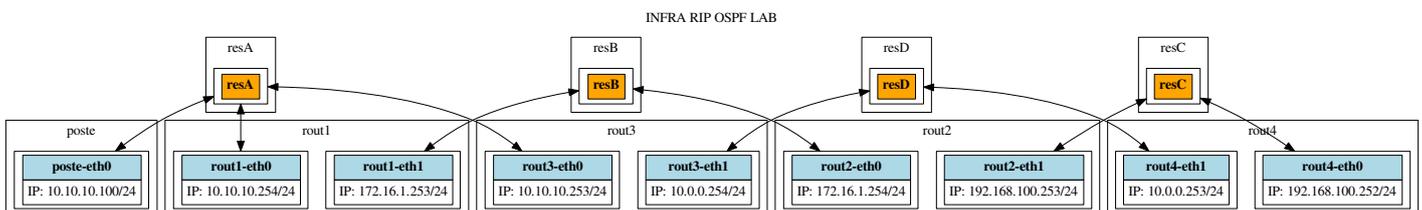


**But de la simulation**

- ▷ étendre le réseau proposé dans la fiche de TP n°1 ;
- ▷ configurer et déployer le protocole RIP dans cette nouvelle configuration ;
- ▷ étudier les paquets échangés du protocole RIP dans cette nouvelle configuration ;
- ▷ générer une panne due à la perte d'une liaison entre routeurs ;
- ▷ analyser les paquets échangés et le fonctionnement du protocole suite à la détection de la panne ;

**Extension de la simulation :**

- \* on ajoute :
  - ◇ un netns : « rout3 » sur lequel tournera le service « frr » ;
  - ◇ un switch que l'on nommera « resD » ;
- \* vous vérifierez que « rout3 » est bien configuré.



## ■■■■ Travail

- vous finaliserez la configuration de la simulation étendue :
  - vous réaliserez la configuration de RIP sur « rout3 » et « rout4 » ;
  - vous vérifierez que les tables de routage de « rout3 » et « rout4 » sont bien configurées ;
- Par quel chemin est accessible le réseau 10.0.0.0/24 depuis l'hôte ? Est-ce normal ?
- Questions sur le fonctionnement de RIP :
  - quel est la table de routage de « rout3 » ?
  - vous snifferez les paquets RIP reçu sur « rout1 » en provenance de « rout3 » :

```
rezo@ishtar:~/$ netns rout1
rezo@ishtar:~/$ [rout1] sudo tcpdump -lnvv -i rout1-eth0 '(udp and port 520)
and host 10.10.10.253'
```

- Existents-ils des différences entre la table de « rout3 » et la table qu'il diffuse vers « rout1 » ? Pourquoi ?
- Vous installerez la commande `traceroute` :

```
rezo@ishtar:~/$ sudo apt install traceroute
```

Vous exécuterez depuis « rout2 », la commande :

```
rezo@ishtar:~/$ netns rout2
rezo@ishtar:~/$ [rout2] traceroute 10.0.0.254
```

Le résultat est-il correct ?

- Identifiez le nom des différentes interfaces et leur connexion pour « rout1 », « rout3 » et « rout4 » ;
- Installez les éléments de surveillance suivants :

- sur « rout2 », vous surveillerez sa configuration de routage :

```
rezo@ishtar:~/$ netns rout2
rezo@ishtar:~/$ [rout2] watch ip route
```

- sur « rout1 », vous continuerez votre surveillance avec `tcpdump` ;
- sur « rout3 », où vous substituerez le nom de l'interface identifiée au paramètre `INTERFACE` :

```
rezo@ishtar:~/$ netns rout3
rezo@ishtar:~/$ [rout3] sudo tcpdump -lnvvX -i INTERFACE '(udp and port 520)
and host 10.0.0.253'
```

- vous créerez un « incident » de routage sur « rout4 » :

```
rezo@ishtar:~/$ netns rout4
rezo@ishtar:~/$ [rout4] sudo ip link set dev rout4-eth1 down
```

Vous exécuterez depuis « rout2 », la commande :

```
rezo@ishtar:~/$ [rout2] traceroute 10.0.0.254
```

Le résultat est-il correct ?

- Que pouvez vous observer sur la capture par `tcpdump` sur « rout3 » lorsque l'interface sur « rout4 » tombe ? Et sur « rout1 » ?
  - Combien de temps faut-il pour que « rout2 » se rende compte du problème ?
- En réactivant l'interface sur « rout4 » :

```
rezo@ishtar:~/$ [rout4] sudo ip l set dev rout4-eth1 up
```

Est-ce que l'hôte mets beaucoup de temps à se mettre à jour ? Pourquoi ?

- Vous lancerez un « ping » depuis « rout2 » :

```
rezo@ishtar:~/$ [rout2] ping 10.0.0.254
```

Vous ferez de nouveau tomber l'interface sur « rout4 ».

Combien de paquets `icmp` sont perdus avant que la route ne soit rétablie ?

## ■■■■ Configuration et étude d'OSPF

Vous stoppez sur chaque netns, le service frr :

▷ sur « rout1 » :

```
rez@ishtar:~/INFRA_LAB$ [rout1] sudo /usr/lib/frr/frrinit.sh stop rout1
```

▷ sur « rout2 » :

```
rez@ishtar:~/INFRA_LAB$ [rout2] sudo /usr/lib/frr/frrinit.sh stop rout2
```

▷ sur « rout3 » :

```
rez@ishtar:~/INFRA_LAB$ [rout3] sudo /usr/lib/frr/frrinit.sh stop rout3
```

▷ sur « rout4 » :

```
rez@ishtar:~/INFRA_LAB$ [rout4] sudo /usr/lib/frr/frrinit.sh stop rout4
```

Vous lancerez le script `init_frr_ospf` qui va configurer le lancement du démon `ospfd` à la place du démon `ripd`.

```
rez@ishtar:~/INFRA_LAB$ ./init_frr_ospf
```

Puis sur chaque netns vous lancerez les démons `zebra` et `ospfd` avec le service frr :

▷ sur « rout1 » :

```
rez@ishtar:~/INFRA_LAB$ [rout1] sudo /usr/lib/frr/frrinit.sh start rout1
```

▷ sur « rout2 » :

```
rez@ishtar:~/INFRA_LAB$ [rout2] sudo /usr/lib/frr/frrinit.sh start rout2
```

▷ sur « rout3 » :

```
rez@ishtar:~/INFRA_LAB$ [rout3] sudo /usr/lib/frr/frrinit.sh start rout3
```

▷ sur « rout4 » :

```
rez@ishtar:~/INFRA_LAB$ [rout4] sudo /usr/lib/frr/frrinit.sh start rout4
```

Vous vérifierez que sur chaque netns `ospfd` fonctionne en attente sur le port 2604 :

```
rez@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=17327,fd=27))
LISTEN    0         3                127.0.0.1:2604      0.0.0.0:*
users: (("ospfd",pid=17332,fd=12))
LISTEN    0         3                127.0.0.1:2616      0.0.0.0:*
users: (("staticd",pid=17335,fd=12))
```

Ensuite, vous pourrez vous connecter pour configurer OSPF :

```
rez@ishtar:~/INFRA_LAB$ [rout1] telnet 127.0.0.1 2604
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:
```

*Le mot de passe est zebra.*

La procédure de configuration pour OSPF est la suivante :

- a. l'ensemble des routeurs va faire partie de l'« area 0 » ou le « backbone » ;
- b. chaque routeur va être identifié par un ID, exprimé sous la forme d'une adresse IP (si on laisse le routeur s'auto-configurer, il prend comme ID l'adresse de son interface interprétable numériquement sur 32 bits comme étant la plus grande) :

- ◇ rout1:1.1.1.1;
- ◇ rout2:2.2.2.2;
- ◇ rout3:3.3.3.3;
- ◇ rout4:4.4.4.4;

Pour configurer l'ID par exemple sur rout1 :

```
xterm
Router> enable
Router# configure terminal
Router(config)# router ospf
Router(config-router)# router-id 1.1.1.1
```

- c. Pour configurer les réseaux à prendre en charge, par exemple sur rout2 :

```
xterm
Router(config-router)# network 172.16.1.0/24 area 0
Router(config-router)# network 192.168.100.0/24 area 0
Router(config-router)# default-information originate
Router(config-router)# write file
Configuration saved to /etc/frr/rout1/ospfd.conf
```

### Étude du fonctionnement d'OSPF :

- ★ Pour vérifier le fonctionnement d'OSPF, par exemple sur rout2 :

```
xterm
Router# show ip ospf
OSPF Routing Process, Router ID: 2.2.2.2
Supports only single TOS (TOS0) routes
This implementation conforms to RFC2328
RFC1583Compatibility flag is disabled
OpaqueCapability flag is disabled
Initial SPF scheduling delay 0 millisecc(s)
Minimum hold time between consecutive SPF's 50 millisecc(s)
Maximum hold time between consecutive SPF's 5000 millisecc(s)
Hold time multiplier is currently 1
SPF algorithm last executed 4m15s ago
Last SPF duration 19 useccs
SPF timer is inactive
LSA minimum interval 5000 msec
LSA minimum arrival 1000 msec
Write Multiplier set to 20
Refresh timer 10 sec
Maximum multiple paths(ECMP) supported 256
This router is an ASBR (injecting external routing information)
Number of external LSA 0. Checksum Sum 0x00000000
Number of opaque AS LSA 0. Checksum Sum 0x00000000
Number of areas attached to this router: 1
Area ID: 0.0.0.0 (Backbone)
  Number of interfaces in this area: Total: 2, Active: 2
  Number of fully adjacent neighbors in this area: 0
  Area has no authentication
  SPF algorithm executed 4 times
  Number of LSA 1
  Number of router LSA 1. Checksum Sum 0x00007b12
  Number of network LSA 0. Checksum Sum 0x00000000
  Number of summary LSA 0. Checksum Sum 0x00000000
  Number of ASBR summary LSA 0. Checksum Sum 0x00000000
  Number of NSSA LSA 0. Checksum Sum 0x00000000
  Number of opaque link LSA 0. Checksum Sum 0x00000000
  Number of opaque area LSA 0. Checksum Sum 0x00000000
```

- ★ Pour voir la gestion des interfaces, par exemple sur « rout3 » :

```

xterm
Router# show ip ospf interface
rout3-eth0 is up
  ifindex 29, MTU 1500 bytes, BW 10000 Mbit <UP,BROADCAST,RUNNING,MULTICAST>
  Internet Address 10.10.10.253/24, Broadcast 10.10.10.255, Area 0.0.0.0
  MTU mismatch detection: enabled
  Router ID 3.3.3.3, Network Type BROADCAST, Cost: 10
  Transmit Delay is 1 sec, State Backup, Priority 1
  Designated Router (ID) 1.1.1.1 Interface Address 10.10.10.254/24
  Backup Designated Router (ID) 3.3.3.3, Interface Address 10.10.10.253
  Multicast group memberships: OSPFAllRouters OSPFDesignatedRouters
  Timer intervals configured, Hello 10s, Dead 40s, Wait 40s, Retransmit 5
    Hello due in 6.636s
  Neighbor Count is 1, Adjacent neighbor count is 1
rout3-eth1 is up
  ifindex 31, MTU 1500 bytes, BW 10000 Mbit <UP,BROADCAST,RUNNING,MULTICAST>
  Internet Address 10.0.0.254/24, Broadcast 10.0.0.255, Area 0.0.0.0
  MTU mismatch detection: enabled
  Router ID 3.3.3.3, Network Type BROADCAST, Cost: 10
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 3.3.3.3 Interface Address 10.0.0.254/24
  Backup Designated Router (ID) 4.4.4.4, Interface Address 10.0.0.253
  Multicast group memberships: OSPFAllRouters OSPFDesignatedRouters
  Timer intervals configured, Hello 10s, Dead 40s, Wait 40s, Retransmit 5
    Hello due in 2.886s
  Neighbor Count is 1, Adjacent neighbor count is 1

```

- ★ Pour voir les relations d'adjacence sur rout3 :

```

xterm
Router# show ip ospf neighbor

```

Neighbor ID	Pri	State	Dead Time	Address	Interface
1.1.1.1	1	Full/DR	35.456s	10.10.10.254	rout3-eth0:10.10.10.253
4.4.4.4	1	Full/Backup	34.472s	10.0.0.253	rout3-eth1:10.0.0.254

- ★ Pour voir les routes, sur « rout3 » :

```

xterm
Router# show ip ospf route
===== OSPF network routing table =====
N   10.0.0.0/24          [10] area: 0.0.0.0
    directly attached to rout3-eth1
N   10.10.10.0/24       [10] area: 0.0.0.0
    directly attached to rout3-eth0
N   10.10.20.0/24       [20] area: 0.0.0.0
    via 10.10.10.254, rout3-eth0
N   172.16.1.0/24       [20] area: 0.0.0.0
    via 10.10.10.254, rout3-eth0
N   192.168.100.0/24    [20] area: 0.0.0.0
    via 10.0.0.253, rout3-eth1

===== OSPF router routing table =====
R   2.2.2.2             [20] area: 0.0.0.0, ASBR
    via 10.0.0.253, rout3-eth1
    via 10.10.10.254, rout3-eth0

===== OSPF external routing table =====

```

- ★ Pour voir la base de donnée d'un routeur :

```
xterm
Router# show ip ospf database

      OSPF Router with ID (4.4.4.4)

      Router Link States (Area 0.0.0.0)

Link ID        ADV Router    Age  Seq#       CkSum  Link count
1.1.1.1        1.1.1.1      406 0x80000008 0xd458 3
2.2.2.2        2.2.2.2      329 0x80000007 0xfc0b 2
3.3.3.3        3.3.3.3      337 0x80000006 0xac1c 2
4.4.4.4        4.4.4.4      328 0x80000006 0x71f2 2

      Net Link States (Area 0.0.0.0)

Link ID        ADV Router    Age  Seq#       CkSum
10.0.0.254     3.3.3.3      336 0x80000001 0xcd5b
10.10.10.254   1.1.1.1      407 0x80000001 0x9c8c
172.16.1.254   2.2.2.2      564 0x80000001 0x4544
192.168.100.253 2.2.2.2      329 0x80000001 0x442a
```

### ■■■■ Travail

1. Quel est le coût d'une liaison par défaut ?
2. Vous snifferez les paquets échangés par OSPF à l'aide de la commande suivante (OSPF est associé au protocole 89 dans le datagramme IP) :

```
xterm
rezo@ishtar:~/ $ netns rout3
rezo@ishtar:~/ $ [rout3] sudo tcpdump -lnv -i rout3-eth0 'ip[9] == 89'
```

3. Sur rout2 vous essaieriez la commande traceroute :

```
xterm
root@ishtar:~/ $ [rout2] traceroute 10.0.0.254
```

Quel est le chemin emprunté ?

Consultez la table de routage, est-ce conforme ?

4. Vous modifierez le coût associé au lien de « rout4 » sur son interface rout4-eth1 :

```
xterm
Router# configure terminal
Router(config)# interface rout4-eth1
Router(config-if)# ospf cost 30
```

Vous ferez de même pour l'interface connectée à « resC » sur rout2.

Pourquoi doit-on faire la modification sur « rout4 » et rout2 ?

Est-ce que le traceroute donne le même résultat ?

La table de routage a-t-elle été modifiée ?

Est-ce conforme à la théorie ?

5. Vous rétablirez les coûts sur rout4 et sur rout2.

En recommandant une surveillance d'affichage de la table de routage de rout2 :

```
xterm
root@ishtar:~/ $ [rout2] watch ip route
```

Vous ferez tomber l'interface rout4-eth1 sur rout4.

```
xterm
root@ishtar:~/ $ [rout4] sudo ip link set dev rout4-eth1 down
```

Est-ce que la modification est rapide ?

Vous réactiverez l'interface et vous lancerez un ping depuis rout2 :

```
xterm
root@ishtar:~/ $ [rout2] ping 10.0.0.254
```

Puis de nouveau de désactiver l'interface : combien de paquets icmp sont perdus ?