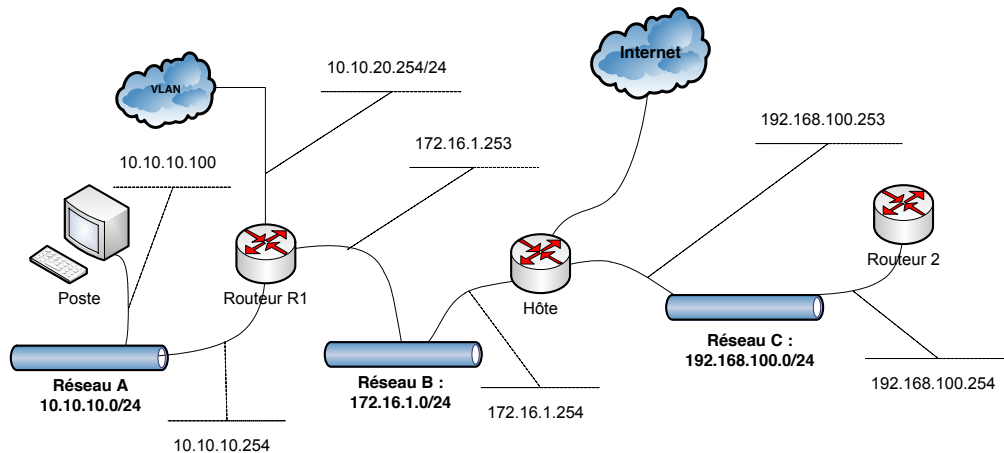


Routing dynamique avec RIP

■ ■ ■ Routing à l'aide de RIP



But de la simulation

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

Travail :

1. Sur chaque poste vérifiez les informations de routage construites ;
2. à l'aide de tcpdump, *sniffez* et étudiez les paquets du protocole RIP sur les différentes interfaces des différents routeurs : `tcpdump -nvv udp`.

```
xterm
root@Routeur1:~/RIP# tcpdump -nvv udp
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 65535 bytes
03:10:41.187109 IP (tos 0xc0, ttl 1, id 0, offset 0, flags [DF], proto UDP (17), length 72)
  172.16.1.253.520 > 224.0.0.9.520: [udp sum ok]
RIPv2, Response, length: 44, routes: 2
  AFI IPv4,      10.10.10.0/24, tag 0x0000, metric: 1, next-hop: self
  AFI IPv4,      10.10.20.0/24, tag 0x0000, metric: 1, next-hop: self
0x0000: 0202 0000 0002 0000 0a0a 0a00 ffff ff00
0x0010: 0000 0000 0000 0001 0002 0000 0a0a 1400
0x0020: ffff ff00 0000 0000 0000 0001 0002 0000
03:10:44.340302 IP (tos 0xc0, ttl 1, id 0, offset 0, flags [DF], proto UDP (17), length 92)
  172.16.1.254.520 > 224.0.0.9.520: [udp sum ok]
RIPv2, Response, length: 64, routes: 3
  AFI IPv4,      0.0.0.0/0, tag 0x0000, metric: 1, next-hop: self
  AFI IPv4,      192.168.100.0/24, tag 0x0000, metric: 1, next-hop: self
  AFI IPv4,      192.168.127.0/24, tag 0x0000, metric: 1, next-hop: self
0x0000: 0202 0000 0002 0000 0000 0000 0000 0000
0x0010: 0000 0000 0000 0001 0002 0000 c0a8 6400
0x0020: ffff ff00 0000 0000 0000 0001 0002 0000
0x0030: c0a8 7f00 ffff ff00 0000 0000 0000 0001
^C
2 packets captured
2 packets received by filter
0 packets dropped by kernel
```

3. à l'aide de NetFilter donnez l'accès Internet (iptables, table « nat », chaîne POSTROUTING etc.)
 - ◇ sur l'hôte : pour les réseaux 172.16.1.0/24 et 192.168.100.0/24 ;
 - ◇ sur « routeur 1 » : pour le réseau 10.10.10.0/24.

```
xterm
sudo iptables -t nat -A POSTROUTING -s 10.10.10.0/24 -j MASQUERADE
sudo iptables -t nat -A POSTROUTING -s 172.16.1.0/24 -j MASQUERADE
sudo iptables -t nat -A POSTROUTING -s 192.168.0.0/16 -j MASQUERADE
```