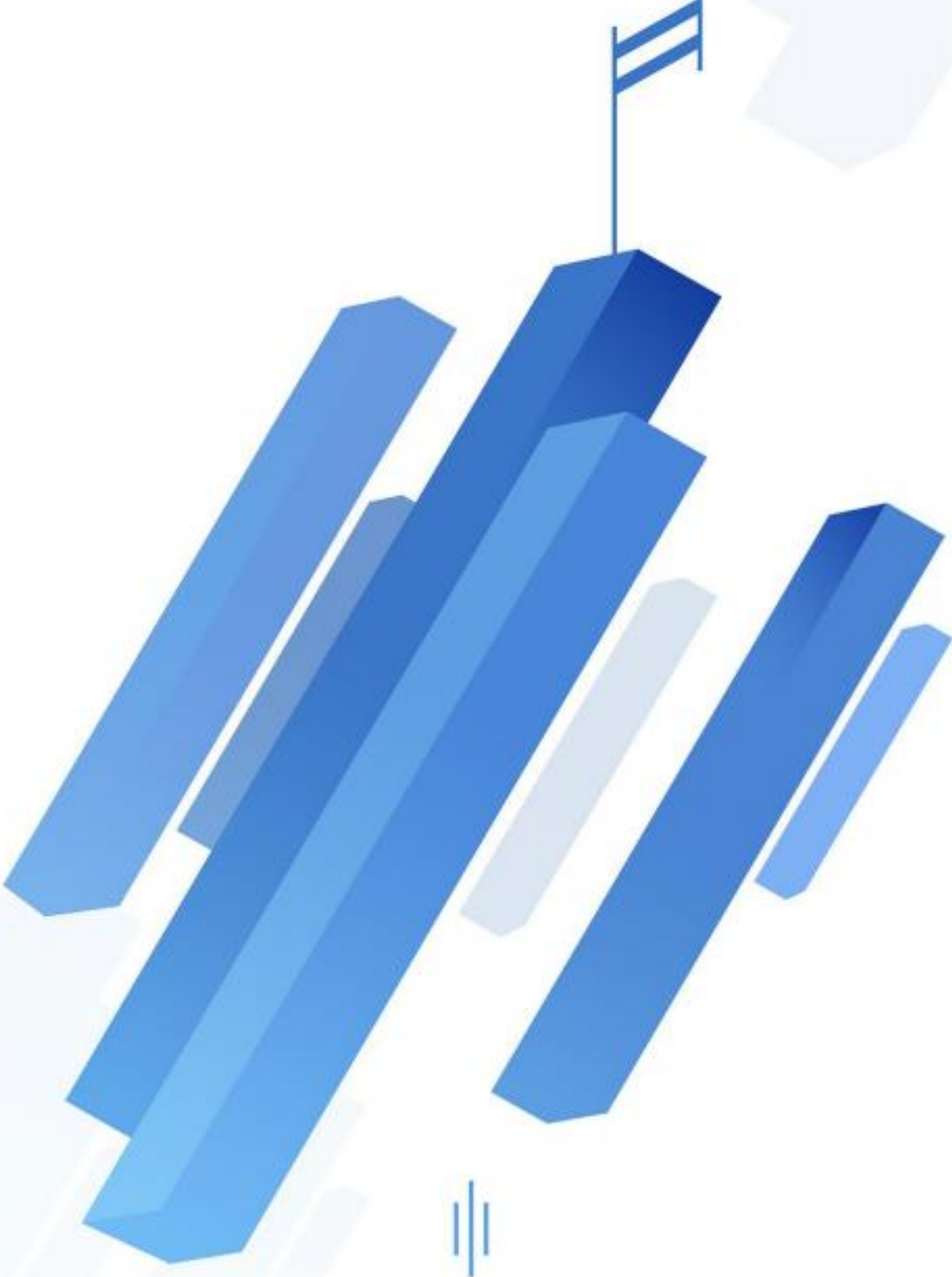


ROUTAGE RIP



Compte-rendu de situation professionnelle : Mise en place du routage dynamique avec RIP v2

Table des matières (à mettre à jour dans Word via clic droit → Mettre à jour le champ)

Cahier des charges – Expression des besoins

Descriptif de l'existant

L'infrastructure repose sur trois routeurs virtuels (R1, R2, R3) déployés sous Debian dans un environnement Proxmox. Chaque routeur dispose de deux interfaces réseau pour assurer l'interconnexion avec les trois clients.

Tableau d'adressage :

Equipement	Interface	Adresse IP	Masque
R1	ens19 (vers R2)	10.1.1.1	/30
R1	ens18 (LAN)	172.30.10.1	/24
R2	ens18 (vers R1)	10.1.1.2	/30
R2	ens19 (vers R3)	10.2.2.2	/30
R2	ens20 (LAN)	209.165.201.1	/24
R3	ens18 (vers R2)	10.2.2.1	/30
R3	ens19 (LAN)	172.30.30.1	/24

[PC-B]

209.165.201.2/24

|

|

209.165.201.1/24

|

[R2]

10.1.1.2/30 | 10.2.2.2/30

-----/\-----

/

\

10.1.1.1/30

10.2.2.1/30

[R1]

[R3]

172.30.10.1/24

172.30.30.1/24

|

|

172.30.10.3/24

172.30.30.3/24

[PC-A]

[PC-C]

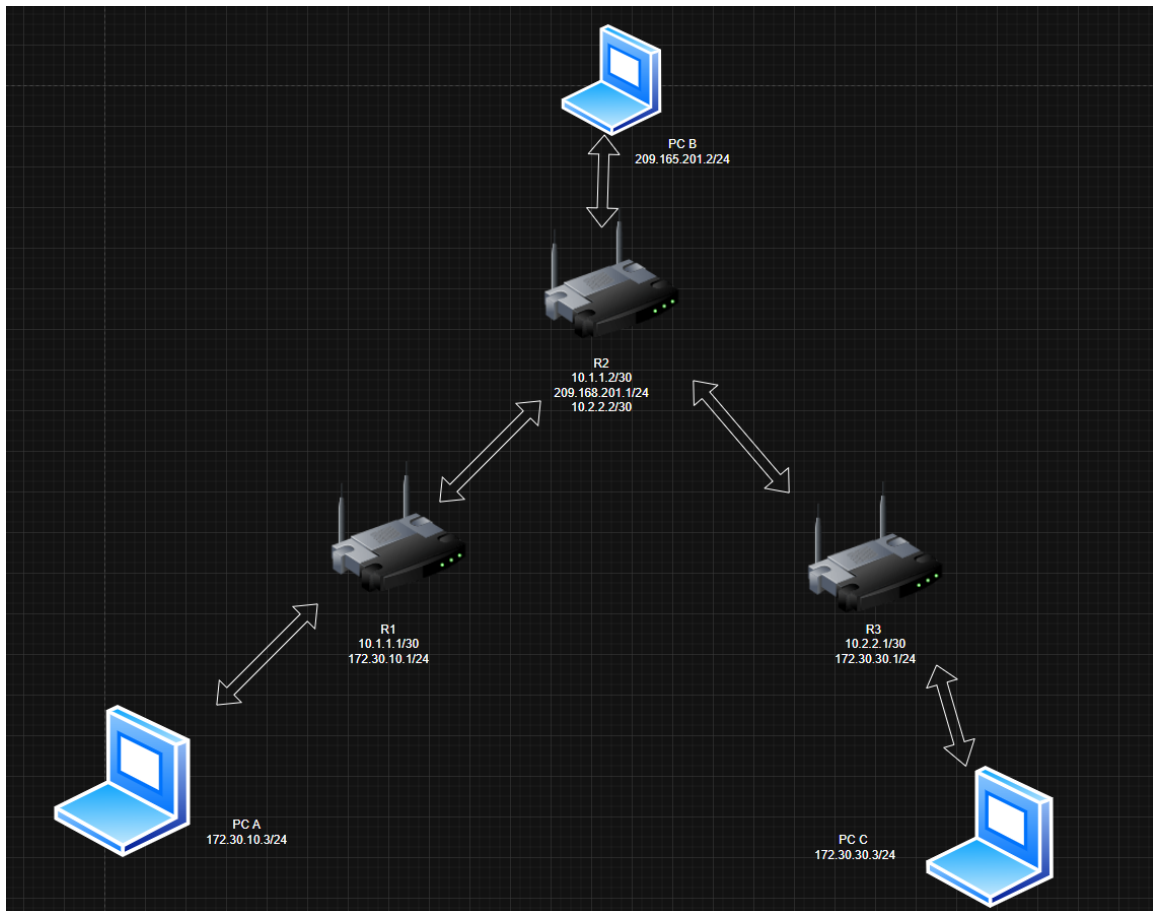


Schéma draw.io de l'infra

Besoins

Assurer l'interconnexion automatique des trois routeurs et de leurs réseaux locaux sans configuration de routes statiques, via un protocole de routage dynamique adapté.

Contraintes

- Temps : déploiement rapide en conditions de TP.
- Budget : utilisation de paquets libres (FRR).
- Organisation : configuration réalisée sur VMs Debian sous Proxmox.
- Juridique : respect des licences open-source (FRR).

Analyse

Solutions envisageables

Deux approches :

- Routage statique : simple mais peu évolutif.
- Routage dynamique : RIP v2

Comparaison

- Statique : maintenance lourde.
- OSPF/BGP : complexes, surdimensionnés ici.
- RIP v2 : adapté aux topologies simples, facile à configurer.

Choix

RIP v2 est retenu pour sa simplicité et son adéquation avec un TP pédagogique.

Impact sur le SI

- Sécurité : acceptable en réseau isolé.
- Performance : convergence rapide pour 3 routeurs.
- Ergonomie : administration simplifiée.
- Juridique : aucun impact.

Tests prévus

- Ping entre interfaces adjacentes.
- Vérification des routes via `show ip route`.
- Ping de bout en bout entre R1 et R3.

R1 :

```
root@debian:~# ip route
10.1.1.0/30 dev ens19 proto kernel scope link src 10.1.1.1
10.2.2.0/30 nhid 12 via 10.1.1.2 dev ens19 proto rip metric 20
172.30.10.0/24 dev ens18 proto kernel scope link src 172.30.10.1
172.30.30.0/24 nhid 12 via 10.1.1.2 dev ens19 proto rip metric 20
209.165.201.0/24 nhid 12 via 10.1.1.2 dev ens19 proto rip metric 20
root@debian:~#
```

```
debian# 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) 10.1.1.0/30    0.0.0.0         1 self           0
R(n) 10.2.2.0/30    10.1.1.2        2 10.1.1.2        0 02:38
C(i) 172.30.10.0/24 0.0.0.0         1 self           0
R(n) 172.30.30.0/24 10.1.1.2        3 10.1.1.2        0 02:38
R(n) 209.165.201.0/24 10.1.1.2        2 10.1.1.2        0 02:38
debian# exit
```

R2:

```
root@debian:~# ip route show
10.1.1.0/30 dev ens18 proto kernel scope link src 10.1.1.2
10.2.2.0/30 dev ens20 proto kernel scope link src 10.2.2.2
172.30.10.0/24 nhid 10 via 10.1.1.1 dev ens18 proto rip metric 20
172.30.30.0/24 nhid 16 via 10.2.2.1 dev ens20 proto rip metric 20
209.165.201.0/24 dev ens19 proto kernel scope link src 209.165.201.1
root@debian:~#
```

```
debian# 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) 10.1.1.0/30       0.0.0.0           1 self           0
C(i) 10.2.2.0/30       0.0.0.0           1 self           0
R(n) 172.30.10.0/24    10.1.1.1          2 10.1.1.1       0 02:43
R(n) 172.30.30.0/24    10.2.2.1          2 10.2.2.1       0 02:45
C(i) 209.165.201.0/24 0.0.0.0           1 self           0
debian# exit
```

R3:

```
root@debian:~# ip route show
10.1.1.0/30 nhid 8 via 10.2.2.2 dev ens18 proto rip metric 20
10.2.2.0/30 dev ens18 proto kernel scope link src 10.2.2.1
172.30.10.0/24 nhid 8 via 10.2.2.2 dev ens18 proto rip metric 20
172.30.30.0/24 dev ens19 proto kernel scope link src 172.30.30.1
209.165.201.0/24 nhid 8 via 10.2.2.2 dev ens18 proto rip metric 20
root@debian:~#
```

```
Hello, this is FRRouting (version 8.4.4).
Copyright 1996-2005 Kunihiro Ishiguro, et al.

debian# 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(n) 10.1.1.0/30       10.2.2.2          2 10.2.2.2       0 02:47
C(i) 10.2.2.0/30       0.0.0.0           1 self           0
R(n) 172.30.10.0/24    10.2.2.2          3 10.2.2.2       0 02:47
C(i) 172.30.30.0/24    0.0.0.0           1 self           0
R(n) 209.165.201.0/24 10.2.2.2          2 10.2.2.2       0 02:47
debian# _
```

Mise en place

Méthodologie

1. Attribution des adresses IP aux interfaces.
2. Installation de FRR.
3. Activation du démon RIP.
4. Configuration des réseaux RIP.
5. Validation par ping et `show ip route`.

Configuration RIP FRR

R1 :

```
router rip
version 2
network 10.1.1.0/30
network 172.30.10.0/24
```

R2 :

```
router rip
version 2
network 10.1.1.0/30
network 10.2.2.0/30
network 209.165.201.0/24
```

R3 :

```
router rip
version 2
network 10.2.2.0/30
network 172.30.30.0/24
```

**ATTENTION A LA FIN DE LA CONFIGURATION RIP IL FAUT BIEN FAIRE LA COMMANDE
« exit » deux fois**

Par la suite on fait « write memory » par default il va créer le fichier conf rip en cas de redémarrage ça enregistre directement tout dans le fichier

Annexes – Commandes utiles

- Installation : `sudo apt install fr`
- Activer RIP : éditer `/etc/fr/daemons` et mettre `ripd=yes`
- Redémarrer FRR : `sudo systemctl restart fr`
- Accéder à FRR : `sudo vtysh`
- Vérifier RIP : `show ip rip`
- Vérifier routes : `show ip route`

PING PC A TO B

```
root@Deb11-Clt:~# ping 209.165.201.2
PING 209.165.201.2 (209.165.201.2) 56(84) bytes of data.
64 bytes from 209.165.201.2: icmp_seq=1 ttl=62 time=0.836 ms
64 bytes from 209.165.201.2: icmp_seq=2 ttl=62 time=0.808 ms
64 bytes from 209.165.201.2: icmp_seq=3 ttl=62 time=0.654 ms
```

PING PC A TO C

```
root@Deb11-Clt:~# ping 172.30.30.3
PING 172.30.30.3 (172.30.30.3) 56(84) bytes of data.
64 bytes from 172.30.30.3: icmp_seq=1 ttl=61 time=0.840 ms
64 bytes from 172.30.30.3: icmp_seq=2 ttl=61 time=0.926 ms
64 bytes from 172.30.30.3: icmp_seq=3 ttl=61 time=0.874 ms
^C
```

PING PC B TO A

```
root@Deb11-Clt:~# ping 172.30.30.3
PING 172.30.30.3 (172.30.30.3) 56(84) bytes of data.
64 bytes from 172.30.30.3: icmp_seq=1 ttl=62 time=0.510 ms
64 bytes from 172.30.30.3: icmp_seq=2 ttl=62 time=0.496 ms
64 bytes from 172.30.30.3: icmp_seq=3 ttl=62 time=0.505 ms
^X64 bytes from 172.30.30.3: icmp_seq=4 ttl=62 time=0.545 ms
64 bytes from 172.30.30.3: icmp_seq=5 ttl=62 time=0.536 ms
64 bytes from 172.30.30.3: icmp_seq=6 ttl=62 time=0.510 ms
^C
```

PING PC B TO C

```
6 packets transmitted, 6 received, 0% packet loss, time 5076ms
rtt min/avg/max/mdev = 0.496/0.517/0.545/0.017 ms
root@Deb11-Clt:~# ping 172.30.30.3
PING 172.30.30.3 (172.30.30.3) 56(84) bytes of data.
64 bytes from 172.30.30.3: icmp_seq=1 ttl=62 time=0.614 ms
64 bytes from 172.30.30.3: icmp_seq=2 ttl=62 time=0.601 ms
64 bytes from 172.30.30.3: icmp_seq=3 ttl=62 time=0.657 ms
^C
```

PING PC C TO B

```
root@Deb11-Clt:~# ping 209.165.201.2
PING 209.165.201.2 (209.165.201.2) 56(84) bytes of data.
64 bytes from 209.165.201.2: icmp_seq=1 ttl=62 time=0.753 ms
64 bytes from 209.165.201.2: icmp_seq=2 ttl=62 time=0.673 ms
64 bytes from 209.165.201.2: icmp_seq=3 ttl=62 time=0.692 ms
^C
```

PING PC C TO A

```
root@Deb11-Clt:~# ping 172.30.10.3
PING 172.30.10.3 (172.30.10.3) 56(84) bytes of data.
64 bytes from 172.30.10.3: icmp_seq=1 ttl=61 time=1.03 ms
64 bytes from 172.30.10.3: icmp_seq=2 ttl=61 time=1.25 ms
64 bytes from 172.30.10.3: icmp_seq=3 ttl=61 time=0.977 ms
^C
```

Bilan

La configuration de RIP v2 avec FRR a permis de valider la diffusion des routes entre les trois routeurs. Les tests de connectivité ont confirmé que les communications de bout en bout sont fonctionnelles.