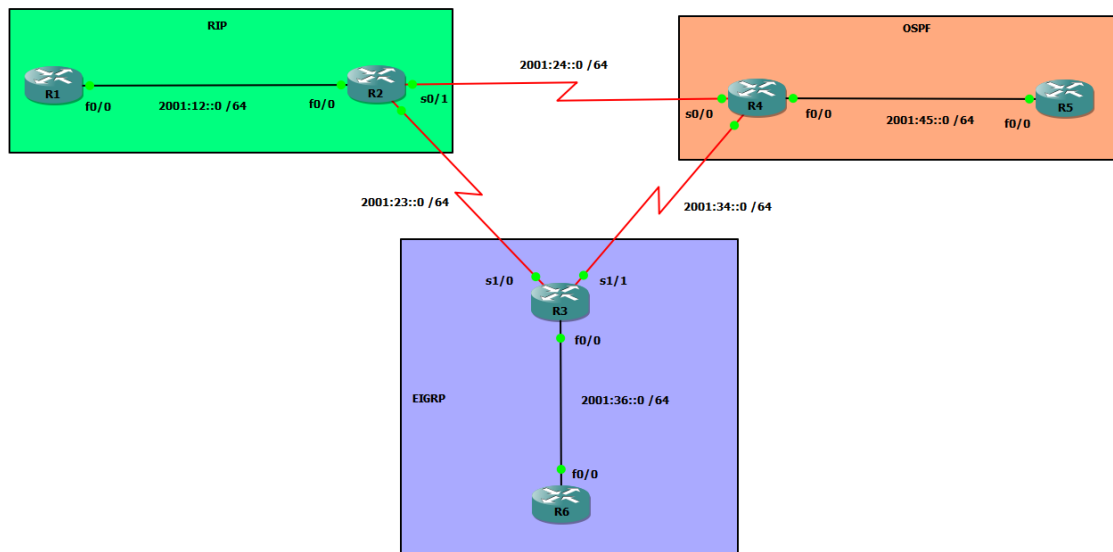


IPv6 : Configuration

1) Présentation de la topologie

Pour ces manipulations, nous allons utiliser une topologie unique.
La voici :



Nous avons trois grands réseaux. Chacun d'eux utilise un protocole de routage différent. Entre R1, R2 et R3, nous utiliserons des routes statiques. Mais avant cela, attardons nous sur la configuration basique, c'est-à-dire les adresses IPv6.

2) Configuration d'interface

Commençons par le plus important : les interfaces !

Cela peut paraître simple, mais en fait il y a plus de choses à maîtriser qu'en IPv4.

Nous avons vu qu'il y a trois types d'adresse :

- **La Local Link**
- **La Global**
- **La Site Local**

La Site Local n'étant plus utilisée, il nous reste les deux autres.

Laquelle allons-nous mettre sur nos interfaces ?

Les deux !

Et oui, chaque interface de notre routeur aura deux IPv6.

La Global pour les communications standards, et la Link Local pour la découverte de voisin, l'échange d'infos pour les protocoles de routage, etc...

Commençons par configurer la Global (en utilisant les IP de la topologie) :

```
R1(config)#interface fastEthernet 0/0
R1(config-if)#ipv6 address 2001:12::1/64
R1(config-if)#no shutdown
```

Pour voir le résultat :

```
R1#show ipv6 interface fastEthernet 0/0
FastEthernet0/0 is administratively down, line protocol is down
  IPv6 is enabled, link-local address is FE80::C000:BFF:FE60:0 [TEN]
  Global unicast address(es):
    2001:12::1, subnet is 2001:12::/64 [TEN]
  Joined group address(es):
    FF02::1
    FF02::1:FF00:1
    FF02::1:FF60:0
  MTU is 1500 bytes
  ICMP error messages limited to one every 100 milliseconds
  ICMP redirects are enabled
  ND DAD is enabled, number of DAD attempts: 1
  ND reachable time is 30000 milliseconds
```

L'IP Global est bien configurée.

Mais vous pouvez voir que l'IP Link Local est aussi configurée.

En fait, elle est auto configurée.

Pour auto configurer son IP, une interface prend le préfix (ici FE80::/64) et y ajoute l'identifiant de l'interface.

Pour obtenir son identifiant unique (EUI-64), l'interface effectue une manipulation à partir de son adresse MAC, afin d'obtenir une adresse de 64 bits.

Pour bien comprendre le processus, voyons comment R1 a pu trouver son IP Link Local au format EUI-64.

Sur la capture précédente, vous pouvez voir le fameux préfix FE80::/64

La deuxième moitié de l'adresse est calculé automatiquement par le routeur.

Pour cela il a besoin de l'adresse MAC de l'interface :

```
R1#show interfaces fa0/0
FastEthernet0/0 is administratively down, line protocol is down
  Hardware is Gt96k FE, address is c200.0b60.0000 (bia c200.0b60.0000)
```

L'adresse MAC est donc : C200.0B60.0000

Il va ensuite ajouter la valeur FFFE au milieu de l'adresse

Ce qui nous donne : C200.0BFF.FE60.0000

Dernière étape : inverser le 7^{ème} bit

Voici le début de l'adresse en binaire (C2) : 11000010

En inversant le 7^{ème} bit, nous obtenons, 11000000, soit C0

L'adresse devient alors C000.0BFF.FE60.0000

En y ajoutant le préfix FE80 ::/64 l'adresse Link Local complète devient :

FE80:0000:0000:0000:C000:0BF:FE60:0000

Cela correspond bien à l'IP donné par le routeur :

```
R1#show ipv6 interface fastEthernet 0/0
FastEthernet0/0 is administratively down, line protocol is down
  IPv6 is enabled, link-local address is FE80::C000:BFF:FE60:0
```

Voici donc comment en IPv6 nous pouvons auto configurer une IP au format EUI-64.
Il faut donc un préfixe de 64 bits, plus l'identifiant de l'interface (le EUI-64).
Il est possible d'auto configurer une IP Link Local, mais aussi une IP Global.

Pour résumer, nous avons configuré une IP Global (sur R1 fa0/0) à la main, et l'IP Link Local s'est auto configurée.

Faisons donc l'inverse sur R2 !

Nous allons configurer l'IP Link Local à la main, et laisser R2 déterminer l'IP Global.

Commençons par l'IP Link Local

```
R2(config-if)#no shutdown
R2(config-if)#ipv6 address FE80::2 link-local
```

Passons à l'IP Global

Étant donné que R2 ne peut pas deviner le préfixe, il faut que R1 l'annonce.

Il faut donc passer les commandes suivantes sur R1 :

```
R1(config)#ipv6 unicast-routing
R1(config)#interface fastEthernet 0/0
R1(config-if)#ipv6 nd prefix 2001:12::/64
```

Maintenant que R1 annonce le préfixe, il ne reste plus qu'à dire à R2 de le récupérer :

```
R2(config-if)#ipv6 address autoconfig
```

Et voici le résultat :

```
R2(config-if)#do show ipv6 int fa0/0
FastEthernet0/0 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::2
  Global unicast address(es):
    2001:12::2, subnet is 2001:12::/64 [PRE]
    valid lifetime 2591994 preferred lifetime 604794
  Joined group address(es):
    FE02::1
    FE02::2
    FE02::1:FF00:2
  MTU is 1500 bytes
  ICMP error messages limited to one every 100 milliseconds
  ICMP redirects are enabled
  ND DAD is enabled, number of DAD attempts: 1
  ND reachable time is 30000 milliseconds
  Default router is FE80::C000:BFF:FE60:0 on FastEthernet0/0
```

Libre à vous de choisir la technique de votre choix pour configurer les IP sur les autres routeurs / interfaces.

Mais avant de passer à la suite, voyons ce qui se passe quand nous configurons une IPv6.

Pour cela, plaçons-nous sur R2 s0/0 et activons le debug ND (Neighbor Discovery) :

```
R2(config)#interface serial 0/0
R2(config-if)#no shutdown
R2(config-if)#do debug ipv6 nd
```

ICMP Neighbor Discovery events debugging is on

Ensuite, configurez l'IP puis observez les Log :

```
R2(config-if)#ipv6 address 2001:23::2/64
R2(config-if)#
*Mar  1 02:52:21.363: ICMPv6-ND: Adding prefix 2001:23::2/64 to Serial0/0
*Mar  1 02:52:22.351: ICMPv6-ND: Sending NS for FE80::C001:BFF:FE60:0 on Serial0/0
R2(config-if)#
*Mar  1 02:52:23.351: ICMPv6-ND: DAD: FE80::C001:BFF:FE60:0 is unique.
*Mar  1 02:52:23.351: ICMPv6-ND: Sending NA for FE80::C001:BFF:FE60:0 on Serial0/0
*Mar  1 02:52:23.351: ICMPv6-ND: Address FE80::C001:BFF:FE60:0/10 is up on Serial0/0
*Mar  1 02:52:23.355: ICMPv6-ND: Sending NS for 2001:23::2 on Serial0/0
R2(config-if)#
*Mar  1 02:52:24.355: ICMPv6-ND: DAD: 2001:23::2 is unique.
*Mar  1 02:52:24.355: ICMPv6-ND: Sending NA for 2001:23::2 on Serial0/0
*Mar  1 02:52:24.355: ICMPv6-ND: Address 2001:23::2/64 is up on Serial0/0
R2(config-if)#
```

Quand une IP est configurée en IPv6, le routeur s'assure que l'IP est unique. Il fait cela pour l'IP Link Local et pour la Global.

Tout d'abord, le routeur envoie un **NS – Neighbor Solicitation**. Cela permet de vérifier si quelqu'un utilise déjà cette IP.

Si personne ne répond, le log **DAD – Detection Address Double** indique que l'IP est libre. Enfin, le routeur envoie un **NA – Neighbor Advertisement** pour indiquer qu'il prend cette IP.

Vous pouvez voir que le même processus est utilisé pour l'IP Global.

Quand vous avez fini, n'oubliez pas désactiver le debug

```
R2#undebug all
```

Si toutes les IP sont configurées, vous pouvez passer à la suite.

3) Le Ping en IPv6

Afin de tester les IP, quoi de mieux qu'un Ping

Pour lancer un Ping, la commande est presque là même :

```
R1#ping ipv6 2001:12::2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:12::2, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 4/3/3
```

Mais que se passe-t-il en coulisse ?

Que se passe-t-il quand on ne connaît pas encore l'adresse MAC du voisin ?

En IPv4, il faut d'abord envoyer une requête ARP.

Et en IPv6 ?

Nous en avons déjà parlé. Le protocole **ND – Neighbor Discovery** est utilisé.

Concrètement, avant de pouvoir lancer un Ping, il faut connaître l'adresse MAC de la destination.

Pour connaître celle-ci, le routeur va envoyer une requête **NS – Neighbor Solicitation** en multicast, sur une adresse à laquelle la destination est abonnée.

Prenons un exemple, ce sera bien plus simple.

Nous souhaitons faire un Ping de R1 vers R2.

R1 a pour IPv6 2001:12::1 et R2 2001:12::2

Nous allons donc envoyer un Ping vers 2001:12::2

C'est ici que NS entre en jeu.

Avant d'envoyer le Ping, R1 va envoyer un NS sur l'IP **FF02::1:FF00:2**

Cette IP est du type Local Link.

Vous pouvez voir que R2 y est abonné :

```
R2#show ipv6 interface fastEthernet 0/0
FastEthernet0/0 is up, line protocol is up
IPv6 is enabled, link-local address is FE80::2
Global unicast address(es):
  2001:12::2, subnet is 2001:12::/64 [PRE]
    valid lifetime 2591862 preferred lifetime 604662
Joined group address(es):
  FF02::1
  FF02::2
  FF02::1:FF00:2
MTU is 1500 bytes
ICMP error messages limited to one every 100 milliseconds
ICMP redirects are enabled
ND DAD is enabled, number of DAD attempts: 1
ND reachable time is 30000 milliseconds
Default router is FE80::C000:BFF:FE60:0 on FastEthernet0/0
```

Mais comment R1 a deviné cette IP ?

Et bien simplement que cette IP respecte un schéma bien précis.

Elle est constituée du préfix FF02 ::/64 suivi de 0001:FFxx:xxxx (où les « x » correspondent aux 24 derniers bits de l'IP Global).

Donc pour résumer, R1 veut envoyer un Ping à R2 sur 2001:12::2

Il commence par envoyer un NS sur FF02::1:FF00:2

R2 va recevoir ce message, et va y répondre (**NA – Neighbor Advertisement**) car il y est abonné.

En répondant, il donne son adresse MAC à R1.

Après quoi, R1 peut envoyer un Ping à R2.

Au passage, si deux routeurs ont les mêmes 24 derniers bits dans leur IP, ils recevront tous les deux le message NS. Par contre, seul le routeur concerné y répondra.

Si vous souhaitez une petite démonstration, voici les étapes :

- Vider la table de voisinage IPv6 : « clear ipv6 neighbors »
- Lancer une capture WireShark : Clic droit sur le lien R1 -> R2, puis « Start Capturing »
- Lancer le Ping : « ping ipv6 2001:12::2 repeat 1 »
- Ouvrir WireShark : Clic Droit sur le lien, « Open WireShark »

38	117.472000	2001:12::1	ff02::1:ff00:2	ICMPV6	86	Neighbor Solicitat
39	117.503000	2001:12::2	2001:12::1	ICMPV6	86	Neighbor Advertise
40	117.518000	2001:12::1	2001:12::2	ICMPV6	114	Echo (ping) reques
41	117.534000	2001:12::2	2001:12::1	ICMPV6	114	Echo (ping) reply

Et voici les 4 étapes d'un Ping (seulement quand le routeur ne connaît pas l'adresse MAC du voisin)

Une fois l'adresse connue, plus besoin de NS et NA.

Il faut donc retenir

NS – Neighbor Solicitation : permet de déterminer l'adresse de niveau 2 d'un routeur voisin, ou de vérifier si l'adresse est utilisée.

NA – Neighbor Advertisement : permet de répondre à un NS, ou d'annoncer un changement d'adresse.

4) Configuration de routes statiques

Il n'est pas nécessaire de tergiverser, les routes statiques n'ont rien de spécial.

Avant de pouvoir effectuer du routage IPv6, il faut entrer la commande suivant sur tous les routeurs

```
R1(config)#ipv6 unicast-routing
```

```
R2(config)#ipv6 unicast-routing
```

Etc...

Pour créer une route statique, rien de plus simple :

```
R2(config)#ipv6 route 2001:36::/64 serial 0/0
```

```
R2(config)#ipv6 route 2001:45::/64 2001:24::4
```

```
R2(config)#ipv6 route 2001:34::/64 serial 0/1
```

Les commandes se passent d'explication !

Faites de même pour R3 et R4 :

```
R3(config)#ipv6 route 2001:12::/64 serial 1/0
```

```
R3(config)#ipv6 route 2001:45::/64 serial 0/1
```

```
R3(config)#ipv6 route 2001:24::/64 serial 0/1
```

```
R4(config)#ipv6 route 2001:12::/64 serial 0/0
```

```
R4(config)#ipv6 route 2001:36::/64 serial 0/1
```

```
R4(config)#ipv6 route 2001:23::/64 serial 0/0
```

Très simple n'est-ce pas ?

5) Configuration de RIPng

RIPng est la version IPv6 de RIP v2.

Il fonctionne de la même manière, possède les mêmes attributs, etc...

Les MAJ sont envoyées en multicast sur l'IP FF02::9 en UDP (port 521)

Pour ce qui est de l'authentification, elle n'est plus gérée par RIP, mais directement par IPv6.

L'activation de RIP se fait directement sur les interfaces :

```
R1(config)#interface fastEthernet 0/0
```

```
R1(config-if)#ipv6 rip IPV6LAB enable
```

```
R2(config)#interface fastEthernet 0/0
```

```
R2(config-if)#ipv6 rip IPV6LAB enable
```

```
R2(config)#interface serial 0/0
```

```
R2(config-if)#ipv6 rip IPV6LAB enable
```

```
R2(config)#interface serial 0/1
```

```
R2(config-if)#ipv6 rip IPV6LAB enable
```

```
R2(config)#ipv6 router rip IPV6LAB
R2(config-rtr)#redistribute static
```

Voyons si cela a fonctionné :

```
R1#show ipv6 route
IPv6 Routing Table - 9 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
       U - Per-user Static route
       I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary
       O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 - OSPF ext 2
       ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
C   2001:12::/64 [0/0]
    via ::, FastEthernet0/0
L   2001:12::1/128 [0/0]
    via ::, FastEthernet0/0
R   2001:23::/64 [120/2]
    via FE80::2, FastEthernet0/0
R   2001:24::/64 [120/2]
    via FE80::2, FastEthernet0/0
R   2001:34::/64 [120/2]
    via FE80::2, FastEthernet0/0
R   2001:36::/64 [120/2]
    via FE80::2, FastEthernet0/0
R   2001:45::/64 [120/2]
    via FE80::2, FastEthernet0/0
L   FE80::/10 [0/0]
    via ::, Null0
L   FF00::/8 [0/0]
    via ::, Null0
```

En effet, R1 connaît toutes les routes !

Un petit Ping pour tester :

```
R1#ping ipv6 2001:34::4
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:34::4, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 28/33/48 ms
```

La commande « Debug ipv6 rip » permet d'observer le fonctionnement des MAJ RIPng :

```
R1#debug ipv6 rip
RIP Routing Protocol debugging is on
R1#
*Mar  1 04:37:57.538: RIPng: Sending multicast update on FastEthernet0/0 for IPV6LAB
*Mar  1 04:37:57.538:          src=FE80::C000:BFF:FE60:0
*Mar  1 04:37:57.538:          dst=FF02::9 (FastEthernet0/0)
*Mar  1 04:37:57.542:          sport=521, dport=521, length=32
*Mar  1 04:37:57.542:          command=2, version=1, mbz=0, #rte=1
*Mar  1 04:37:57.542:          tag=0, metric=1, prefix=2001:12::/64
```

Vous pouvez constater que ce sont les IP Link Local qui sont utilisées pour l'envoi de MAJ. Quant à la version du protocole, il ne s'agit pas de RIP v1, mais de RIPng v1.

6) Configuration d'OSPF V3

La version 3 d'OSPF apporte quelques changements par rapport à la version 2.

Nous laisserons la théorie de côté pour cette fois.

Concentrons-nous sur la pratique, qui elle n'a rien de difficile.

```
R5(config)#ipv6 router ospf 1
R5(config-rtr)#router-id 5.5.5.5
```

```
R5(config)#interface fastEthernet 0/0
R5(config-if)#ipv6 ospf 1 area 0
```

```
R4(config)#ipv6 router ospf 1
R4(config-rtr)#router-id 4.4.4.4
R4(config-rtr)#redistribute static
R4(config-rtr)#passive-interface serial 0/0
R4(config-rtr)#passive-interface serial 0/1
```

```
R4(config)#interface fastEthernet 0/0
R4(config-if)#ipv6 ospf 1 area 0
R4(config)#interface serial 0/0
R4(config-if)#ipv6 ospf 1 area 0
R4(config)#interface serial 0/1
R4(config-if)#ipv6 ospf 1 area 0
```

Voyons la table de routage

```
R5#show ipv6 route
IPv6 Routing Table - 9 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
       U - Per-user Static route
       I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary
       O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 - OSPF ext 2
       ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
OE2 2001:12::/64 [110/20]
    via FE80::C003:3FF:FE38:0, FastEthernet0/0
OE2 2001:23::/64 [110/20]
    via FE80::C003:3FF:FE38:0, FastEthernet0/0
O 2001:24::/64 [110/74]
  via FE80::C003:3FF:FE38:0, FastEthernet0/0
O 2001:34::/64 [110/74]
  via FE80::C003:3FF:FE38:0, FastEthernet0/0
OE2 2001:36::/64 [110/20]
    via FE80::C003:3FF:FE38:0, FastEthernet0/0
C 2001:45::/64 [0/0]
  via ::, FastEthernet0/0
L 2001:45::5/128 [0/0]
  via ::, FastEthernet0/0
L FE80::/10 [0/0]
  via ::, Null0
L FF00::/8 [0/0]
  via ::, Null0
```

Pour le reste, les commandes sont très semblables à la version 2.

7) Configuration d'EIGRP en IPv6

Finissons par une configuration basique d'EIGRP en IPv6.

Voyons quelques commandes de bases pour le mettre en place.

Premièrement, il faut activer le processus EIGRP :

```
R3(config)#ipv6 router eigrp 1
R3(config-rtr)#eigrp router-id 3.3.3.3 (l'ID est obligatoire si aucune IPV4
n'est configurée sur le routeur)
R3(config-rtr)#passive-interface serial 1/0
R3(config-rtr)#passive-interface serial 1/1
R3(config-rtr)#redistribute static
R3(config)#interface serial 1/0
R3(config-if)#ipv6 eigrp 1
R3(config)#interface serial 1/1
R3(config-if)#ipv6 eigrp 1
R3(config)#interface fastEthernet 0/0
R3(config-if)#ipv6 eigrp 1
```

```
R6(config)#ipv6 router eigrp 1
R6(config-rtr)#eigrp router-id 6.6.6.6
R6(config)#interface fastEthernet 0/0
R6(config-if)#ipv6 eigrp 1
```

Constatons le résultat :

```
R6#show ipv6 route
IPv6 Routing Table - default - 8 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
       B - BGP, M - MIPv6, R - RIP, I1 - ISIS L1
       I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary, D - EIGRP
       EX - EIGRP external, ND - Neighbor Discovery
       O - OSPF Intra, OI - OSPF Inter, OE1 - OSPF ext 1, OE2 - OSPF ext 2
       ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
EX 2001:12::/64 [170/2172416]
    via FE80::C809:16FF:FEC4:0, FastEthernet0/0
D 2001:23::/64 [90/2172416]
    via FE80::C809:16FF:FEC4:0, FastEthernet0/0
EX 2001:24::/64 [170/2172416]
    via FE80::C809:16FF:FEC4:0, FastEthernet0/0
D 2001:34::/64 [90/2172416]
    via FE80::C809:16FF:FEC4:0, FastEthernet0/0
C 2001:36::/64 [0/0]
    via FastEthernet0/0, directly connected
L 2001:36::6/128 [0/0]
    via FastEthernet0/0, receive
EX 2001:45::/64 [170/2172416]
    via FE80::C809:16FF:FEC4:0, FastEthernet0/0
L FF00::/8 [0/0]
    via Null0, receive
```