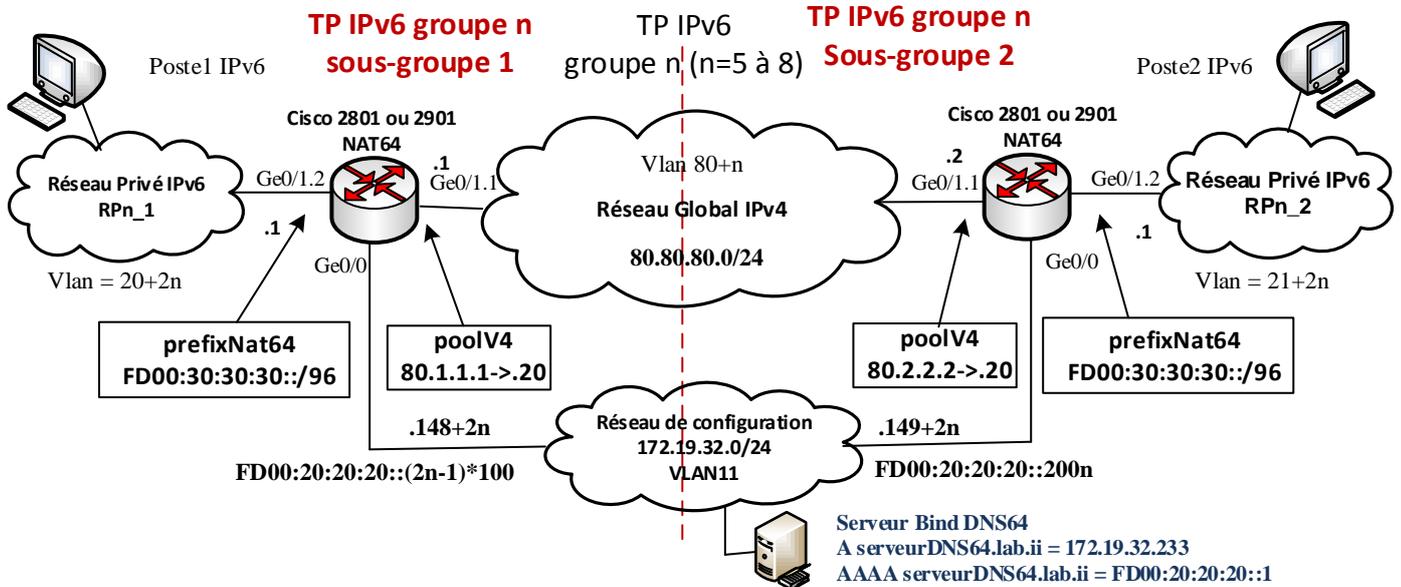


# Travaux Pratiques IPV6

## Manip 5 : Configuration NAT64-DNS64

Les routeurs Cisco 2901 serviront de routeur NAT64 statefull. Dans un premier temps la NAT64 sera mise en place et testée sans utilisation du DNS64.



### Configuration initiale

- 1) Sur chaque routeur, configurer les interfaces globales IPv4 puis tester la liaison IPv4 entre les deux routeurs.
- 2) Sur chaque routeur, configurer l'interface dans le réseau IPv6 en mode configuration automatique avec état (Eq. Manip1 questions 13 et 14).

Groupe n - sous-groupe 1	Groupe n - sous-groupe 2
<pre> <b>ipv6 dhcp pool poolR1-DHCPv6</b> <b>address prefix FD00:2n-1:2n-1:2n-1::/64 lifetime</b> <b>1800 1500</b> <b>domain-name lab.ii</b> <b>!</b> <b>interface GigabitEthernet0/1.2</b> <b>encapsulation dot1Q 20+2n</b> <b>nat64 enable</b> <b>ipv6 address FD00:2n-1:2n-1:2n-1::1/64</b> <b>ipv6 enable</b> <b>ipv6 nd ns-interval 10000</b> <b>ipv6 nd prefix FD00:2n-1:2n-1:2n-1::/64 no-advertise</b> <b>ipv6 nd managed-config-flag</b> <b>ipv6 nd ra interval 15</b> <b>ipv6 dhcp server poolR1-DHCPv6</b> </pre>	<pre> <b>ipv6 dhcp pool poolR2-DHCPv6</b> <b>address prefix FD00:2n:2n:2n::/64 lifetime 1800</b> <b>1500</b> <b>domain-name lab.ii</b> <b>!</b> <b>interface GigabitEthernet0/1.2</b> <b>encapsulation dot1Q 21+2n</b> <b>nat64 enable</b> <b>ipv6 address FD00:2n:2n:2n::1/64</b> <b>ipv6 enable</b> <b>ipv6 nd ns-interval 10000</b> <b>ipv6 nd prefix FD00:2n:2n:2n::/64 no-advertise</b> <b>ipv6 nd managed-config-flag</b> <b>ipv6 nd ra interval 15</b> <b>ipv6 dhcp server poolR2-DHCPv6</b> </pre>

- 3) Vérifier la bonne distribution des adresses IPv6 sur les postes des réseaux privés IPv6.

## Configuration de la NAT64

Pour la configuration de la NAT64 sur le routeur du groupe n sous-groupe 1 on envisage de configurer :

- ✓ une liste d'accès ipv6 nommée **nat64Liste** qui autorisera les adresses source IPv6 **FD00:2n-1:2n-1:2n-1::/64** du réseau privé IPv6 à être « nattées » quelle que soit la destination,
- ✓ un préfixe de NAT64 avec état égal à **FD00:30:30:30::/96** auquel viendra s'ajouter les 32bits des adresses IPv4 de destination,
- ✓ un pool d'adresses IPv4 pour la translation d'adresses v6->v4 nommé **poolNAT64** comprenant les adresses **80.1.1.1** à **80.1.1.20** avec translation de port (overload).

De la même façon, pour la configuration de la NAT64 sur le routeur du groupe n sous-groupe 2 on envisage de configurer :

- ✓ une liste d'accès ipv6 nommée **nat64Liste** qui autorisera les adresses source IPv6 **FD00:2n:2n:2n::/64** du réseau privé IPv6 à être « nattées » quelle que soit la destination,
- ✓ un préfixe de NAT64 avec état égal à **FD00:30:30:30::/96** auquel viendra s'ajouter les 32bits des adresses IPv4 de destination,
- ✓ un pool d'adresses IPv4 pour la translation d'adresses v6->v4 nommé **poolNAT64** comprenant les adresses **80.2.2.1** à **80.2.2.20** avec translation de port (overload).

- 4) Configurer la liste d'accès **nat64Liste**, commande **ipv6 access-list**.
- 5) Configurer le préfix IPv6 de la NAT64 statefull, commande **nat64 prefix stateful**.
- 6) Configurer la NAT64 avec la liste d'accès **nat64Liste** et le pool d'adresses IPv4 **poolNAT64** pour la translation v6->v4 avec translation de port, commandes **nat64 v4 pool ...**, puis **nat64 v6v4 list ... pool ...**
- 7) Activer le service de NAT64 sur les interfaces côté IPv6 et côté IPv4, commande **nat64 enable**.
- 8) Tester le fonctionnement de la NAT 64 en « pingant » l'adresse IPv4 du routeur opposé depuis votre réseau privé IPv6, par exemple depuis le réseau RPN\_1 l'adresse préfixée NAT64 du routeur RPN\_2 sera **FD00:30:30:30::5050:5002** ou **FD00:30:30:30::80.80.80.2**. Activer le mode débogage de la NAT64 pour visualiser les translations réalisées.

Commandes utiles :

- ✓ **debug nat64 translation detail**
- ✓ **show nat64 translation**

## Configuration de la NAT64-DNS64

- 9) Ajouter à la configuration des serveurs DHCP la distribution de l'adresse du serveur DNS **FD00:20:20:20::1** pour que les clients IPv6 aient connaissance du serveur DNS assurant également la fonction de DNS64.
- 10) Afin de pouvoir joindre le serveur DNS en IPv6, activer IPv6 sur les interfaces du réseau de configuration et ajouter sur l'interface Ge0/0 l'adresse IPv6 spécifiée sur le schéma, **FD00:20:20:20::(2n-1)\*100** ou **FD00:20:20:20::2n\*100** selon le routeur.
- 11) Vérifier que les postes client des réseaux privés IPv6 RPN\_1 et RPN\_2 sont bien configurés en simple pile IPv6 et qu'ils ont connaissance du serveur DNS.

Le serveur DNS64 en charge du domaine lab.ii dispose des entrées listées dans le fichier **lab.ii.zone** (en annexe) pour résoudre les adresses des routeurs. En voici un extrait :

- routeur1v4.lab.ii = 80.80.80.1,
- routeur2v4.lab.ii = 80.80.80.2,
- routeur1gr1v6.lab.ii = FD00:1:1:1::1,
- routeur2gr1v6.lab.ii = FD00:2:2:2::1,

- Etc...

- 12) Depuis le poste client réaliser un ping vers l'adresse du routeur opposé dans le réseau global afin de vérifier l'obtention de l'adresse IPv6 préfixée NAT64 par le serveur DNS64 permettant de joindre celui-ci via la NAT64. Vous pouvez également capturer les paquets associés.
- 13) Les accès au réseau global IPv4 depuis les réseaux privés IPv6 sont à présent fonctionnels, mais il est impossible de communiquer entre les réseaux privés IPv6 via le réseau global IPv4. Nous avons résolu ce type de situation dans la manipulation 2 en mettant en place un tunnel 6to4. Faire de même ici afin de permettre la communication entre les réseaux privés IPv6.
- 14) Sauvegarder votre configuration dans la mémoire flash du routeur en la nommant **configIPV6-5.txt**.

*Pour information vous trouverez la configuration du serveur DNS (bind9 sur Ubuntu) en annexe.*

## *Annexe :*

### **Configuration du serveur Bind en serveur DNS maître et DNS64 pour le domaine lab.ii**

---

#### *Configuration de l'interface eth0, adresses IP statiques IPv4 et IPv6 :*

fichier /etc/network/interfaces

```
auto eth0
iface eth0 inet static
address 172.19.32.233
netmask 255.255.255.0
network 172.19.32.0
broadcast 172.19.32.255
gateway 172.19.32.254
dns-nameservers 127.0.0.1
```

```
iface eth0 inet6 static
address FD00:20:20:20::1
netmask 64
```

---

#### *Configuration du serveur DNS pour les résolutions DNS, lui-même :*

fichier /etc/resolv.conf

```
nameserver 127.0.0.1
```

---

#### *Configuration des routes statiques vers les réseaux FD00:1:1:1::/64 et FD00:2:2:2::/64 , pour répondre aux requêtes des clients IPv6 :*

fichier /etc/rc.local

```
/sbin/ip -6 route add FD00:1:1:1::/64 via FD00:20:20:20::100 dev eth0
/sbin/ip -6 route add FD00:2:2:2::/64 via FD00: 20:20:20::200 dev eth0
/sbin/ip -6 route add FD00:3:3:3::/64 via FD00: 20:20:20::300 dev eth0
/sbin/ip -6 route add FD00:4:4:4::/64 via FD00: 20:20:20::400 dev eth0
/sbin/ip -6 route add FD00:5:5:5::/64 via FD00: 20:20:20::500 dev eth0
/sbin/ip -6 route add FD00:6:6:6::/64 via FD00: 20:20:20::600 dev eth0
/sbin/ip -6 route add FD00:7:7:7::/64 via FD00: 20:20:20::700 dev eth0
/sbin/ip -6 route add FD00:8:8:8::/64 via FD00: 20:20:20::800 dev eth0
/sbin/ip -6 route add FD00:9:9:9::/64 via FD00: 20:20:20::900 dev eth0
/sbin/ip -6 route add FD00:10:10:10::/64 via FD00: 20:20:20::1000 dev eth0
/sbin/ip -6 route add FD00:11:11:11::/64 via FD00: 20:20:20::1100 dev eth0
/sbin/ip -6 route add FD00:12:12:12::/64 via FD00: 20:20:20::1200 dev eth0
/sbin/ip -6 route add FD00:13:13:13::/64 via FD00: 20:20:20::1300 dev eth0
/sbin/ip -6 route add FD00:14:14:14::/64 via FD00: 20:20:20::1400 dev eth0
/sbin/ip -6 route add FD00:15:15:15::/64 via FD00: 20:20:20::1500 dev eth0
/sbin/ip -6 route add FD00:16:16:16::/64 via FD00: 20:20:20::1600 dev eth0
```

---

#### *Configuration du serveur bind, zone lab.ii pour la résolution directe (zone lab.ii) et inverse (zone 32.19.172.in-addr.arpa) :*

fichier /etc/bind/named.conf.local

```
zone "lab.ii" IN {
    type master;
    file "/etc/bind/lab.ii.zone";
};

zone "32.19.172.in-addr.arpa" {
    type master;
    file "/etc/bind/32.19.172.in-addr.arpa.zone";
};
```

---

#### *Configuration du fichier de zone directe lab.ii.zone :*

fichier /etc/bind/lab.ii.zone

\$ttl 24H

```
lab.ii. IN      SOA      serveurDNS64.lab.ii.  Michel@michel-marie.net. (
    1          ; Serial
    604800     ; Refress
    86400      ; Retry
    2419200    ; Expire
    604800 )    ; Negative cache TTL
;
@          IN      NS       serveurDNS64.lab.ii.
serveurDNS64 IN     A       172.19.32.233
routeur1v4 IN     A       80.80.80.1
routeur2v4 IN     A       80.80.80.2
routeur1gr1v6 IN   AAAA    FD00:1:1:1::1
routeur2gr1v6 IN   AAAA    FD00:2:2:2::1
routeur1gr2v6 IN   AAAA    FD00:3:3:3::1
routeur2gr2v6 IN   AAAA    FD00:4:4:4::1
routeur1gr3v6 IN   AAAA    FD00:5:5:5::1
routeur2gr3v6 IN   AAAA    FD00:6:6:6::1
routeur1gr4v6 IN   AAAA    FD00:7:7:7::1
routeur2gr4v6 IN   AAAA    FD00:8:8:8::1
routeur1gr5v6 IN   AAAA    FD00:9:9:9::1
routeur2gr5v6 IN   AAAA    FD00:10:10:10::1
routeur1gr6v6 IN   AAAA    FD00:11:11:11::1
routeur2gr6v6 IN   AAAA    FD00:12:12:12::1
routeur1gr7v6 IN   AAAA    FD00:13:13:13::1
routeur2gr7v6 IN   AAAA    FD00:14:14:14::1
routeur1gr8v6 IN   AAAA    FD00:15:15:15::1
routeur2gr8v6 IN   AAAA    FD00:16:16:16::1
serveurDNS64 IN   AAAA    FD00:20:20:20::1
```

---

### Configuration du fichier de zone inverse 32.19.172.in-addr.arpa.zone :

fichier /etc/bind/32.19.172.in-addr.arpa.zone

\$ttl 604800

```
@      IN      SOA      serveurDNS64.lab.ii  Michel@michel-marie.net. (
    1          ; Serial
    604800     ; Refress
    86400      ; Retry
    2419200    ; Expire
    604800 )    ; Negative cache TTL
;
@          IN      NS       serveurDNS64.lab.ii.
233       IN      PTR      serveurDNS64.lab.ii.
```

---

### Configuration du fichier d'options pour l'activation du service dns64 et du transfert des requêtes DNS non résolues (domaine non lab.ii) :

fichier /etc/bind/named.conf.options

```
options {
    recursion yes;
    allow-query { any; };

    forwarders {
        172.19.32.2;
        172.19.32.5;
    };

    dns64 FD00:30:30:30::/96 {
        suffix ::;
    };

    Listen-on-v6 { any; };
};
```

