

Réseaux IP

TD 3 - Adressage IPv6 - Interconnexion de réseaux IPv6

1 Adressage IPv6

1. Quels sont les rôles d'une adresse IP ?

Une adresse IP a deux rôles : celui de localiser et d'identifier une interface réseau.

2. Comment peut-on distinguer chacun des rôles d'une adresse IP ?

Les bits de poids fort d'une adresse constituent le localisateur alors que les bits de poids faible constituent l'identifiant d'interface. La frontière est indiquée par la longueur du préfixe (dans la notation CIDR) qui précise le nombre de bits appartenant au localisateur.

3. Combien d'adresses sont attribuables avec le préfixe $2001::/16$?

Avec un $/16$, il reste $128 - 16 = 112$ bits pour l'identificateur soit 2^{112} adresses.

4. Compressez l'écriture de l'adresse $2001:0000:0000:0000:0abc:0000:0000:4321$

$2001::abc:0:0:4321$

5. Quel est le réseau d'appartenance des adresses :

— $2001:88:1f80::203:ffff:4c18:1009/64$

$2001:88:1f80::/64$

— $2001:bb76:7878:2::42/56$

$2001:bb76:7878::/56$

6. Pour chaque couple adresse IP / préfixe présenté ci-dessous, indiquez si on peut l'utiliser dans la configuration d'un poste (client ou serveur) destiné à être joignable sur Internet. Pour chaque cas négatif, veuillez indiquer la raison.

— $2001:660:4701:1001::1/60$

Préfixe dans $2000::/3$ donc utilisable sur un poste destiné à être joignable sur Internet.

— $2001:660:4701:1001::abcd:0:1/64$

Préfixe dans $2000::/3$ donc utilisable sur un poste destiné à être joignable sur Internet.

— $fe80::abcd:abcd:abcd:abcd/10$

Préfixe dans $fe80::/10$ donc il s'agit d'une adresse lien local qui n'est pas routable sur Internet. Elle ne peut donc pas être utilisée pour configurer un poste destiné à être joignable sur Internet.

7. Dans le protocole IPv4, les adresses ont une taille de 32 bits. Indépendamment des préfixes réseaux, en allouant une adresse IPv4 toutes les nanosecondes, combien faut-il de temps pour épuiser un tel espace d'adressage ? Même question en allouant un million d'adresses IPv6 chaque picoseconde (10^{-12} sec.) ?

Dans IPv4, en considérant un adressage plat, il y a 2^{32} adresses différentes. En allouant une adresse toutes les 0,000000001 secondes, on aura épuisé l'espace en $0,000000001 * 2^{32} = 4,2$ secondes. Dans le cas d'IPv6, il faudra $0,000000000001 * 2^{128} / 1000000$ secondes, soit plus de 10 mille milliards d'années pour épuiser l'espace d'adressage.

2 Plan d'adressage

Une micro-université obtient le préfixe $2001:db8:1::/48$ et dispose de deux routeurs nommés *I* et *R*. Le routeur *I* fait l'interface avec Internet. Le routeur *R* sert à découper le réseau interne. L'architecture du réseau est illustrée sur la figure 1.

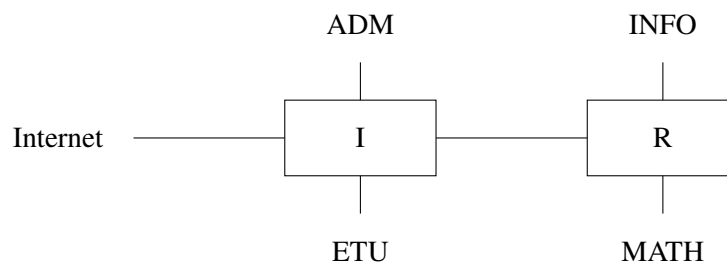


FIGURE 1 – Plan du réseau

1. Choisissez un préfixe de 64 bits pour chaque réseau *ADM*, *ETU*, *INFO* et *MATH*. N'oubliez pas de prévoir également un préfixe au plus juste pour le réseau IP entre les routeurs *I* et *R*.

```

ADM   : 2001:db8:1:0::/64
ETU   : 2001:db8:1:4::/64
INFO  : 2001:db8:1:1::/64
MATH  : 2001:db8:1:2::/64
I-R   : 2001:db8:1:3::/127
  
```

2. Écrivez les tables de routage des routeurs *I* et *R*.

Routeur	destination	prochain saut
I	2001:db8:1:0::/64	local
	2001:db8:1:4::/64	local
	2001:db8:1:3::/127	local
	2001:db8:1:1::/64	R
	2001:db8:1:2::/64	R
R	2001:db8:1:1::/64	local
	2001:db8:1:2::/64	local
	2001:db8:1:3::/127	local
	default	I

3. Exprimez l'espace d'adressage encore disponible pour créer des nouveaux réseaux.

Avec un /48 on peut faire 16 384 (2^{62-48}) /62. On a utilisé un /62 complet pour créer les réseaux *ADM*, *INFO*, *ETU* et *MATH*. Il reste donc 16 383 /62 dans lesquels on a également déjà pris un /127. Avec un /62 on peut créer $2^{65} / 127$. Donc il reste 16 382 /62 et $2^{65} - 1 / 127$.

4. On décide maintenant de découper le réseau *ETU* en trois réseaux nommés *ETU*, *TP1* et *TP2*. Pour cela on ajoute un nouveau routeur *E*.

- (a) Attribuez de nouveaux préfixes pour les réseaux *ETU*, *TP1* et *TP2*.

```
ETU : 2001:db8:1:4::/64
TP1 : 2001:db8:1:5::/64
TP2 : 2001:db8:1:6::/64
```

- (b) Écrivez la table de routage du routeur *E*.

Routeur	destination	prochain saut
E	2001:db8:1:4::/64	local
	2001:db8:1:5::/64	local
	2001:db8:1:6::/64	local
	default	I

- (c) Comment faut-il modifier les tables de routage des routeurs *I* et *R* pour tenir compte de ces nouveaux réseaux ? Écrivez les nouvelles tables de routage si nécessaire.

Sur le routeur *I*, il suffit de remplacer la ligne correspondant au préfixe 2001:db8:1:4::/64 :

Routeur	destination	prochain saut
I	2001:db8:1:0::/64	local
	2001:db8:1:4::/62	E
	2001:db8:1:3::/127	local
	2001:db8:1:1::/64	R
	2001:db8:1:2::/64	R

5. On décide de remplacer le routeur *R* par un commutateur. Détaillez toutes les modifications qu'il faut réaliser sur les autres routeurs et les stations pour tenir compte d'une telle modification.

Sur *E* il n'y a rien à faire. On peut choisir de conserver les préfixes attribués à *INFO* et *MATH* sur le nouveau réseau. Pour ce faire, il faut que *I* les annonce dans ses messages *router advertisements*. Les stations se trouvant à l'origine dans *INFO* (respectivement dans *MATH*) auront par conséquent deux adresses IPv6 globales. Si l'on souhaite conserver la segmentation entre *INFO* et *MATH* malgré le commutateur, on peut utiliser des réseaux locaux virtuels (VLAN) (au programme du S6).

3 Auto-configuration

1. Quelles sont les adresses lien-local attribuées automatiquement aux stations dont les adresses MAC sont :

— 00:03:ff:18:cf:1e

```
fe80::203:ffff:fe18:cf1e
```

— 02:00:4c:4f:4f:50

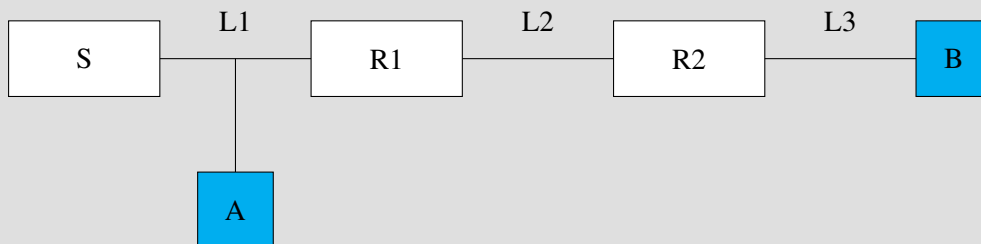
```
fe80::200:4cff:fe4f:4f50
```

2. En considérant les adresses MAC précédentes, quelles seront les adresses globales des stations si elles sont localisées dans le réseau 2001:5a8:cafe:7::/64 ?

4 Résolution d'adresses

On dispose de trois réseaux locaux Ethernet notés L_1 , L_2 et L_3 . Le routeur R_1 connecte L_1 et L_2 . Le routeur R_2 connecte L_2 et L_3 . Le routeur S placé dans L_1 réalise la connexion avec le reste d'Internet. Une station A est placée dans L_1 . Une autre station B est placée dans L_3 . Il n'y a pas d'autre interconnexion.

1. Faites un schéma de l'architecture complète du réseau.



2. Découpez le préfixe `2001:db8:beef::/48` pour attribuer un préfixe à chaque réseau.

On peut réaliser (par exemple) le découpage suivant :

```
L1 - 2001:db8:beef:0::/64
L2 - 2001:db8:beef:1::/64
L3 - 2001:db8:beef:2::/64
```

3. On décide d'utiliser l'auto-configuration sans état (*StateLess Address Autoconfiguration - SLAAC*) pour la configuration IP des postes. En supposant qu'on démarre toutes les stations en même temps, faites la liste de tous les messages échangés pour cette phase.

On suppose que l'adresse MAC de la station A est `00:01:23:45:67:89`. L'auto-configuration sans état débute par la création d'une adresse lien-local en combinant le préfixe `fe80::/10` avec l'EUI-64 associé à l'adresse MAC (ici `201:23ff:fe45:6789`) ce qui donne l'adresse : `fe80::201:23ff:fe45:6789`

Ensuite, la station A doit vérifier l'unicité de cette adresse en réalisant un DAD (*Duplicate Address Detection*). Pour ce faire elle doit en premier lieu s'abonner au groupe multicast correspondant à l'adresse du noeud sollicité, c'est-à-dire l'adresse `ff02::1:ff45:6789` (on combine le préfixe `ff02::1:ff00:0/104` avec les 24 derniers bits de l'adresse que l'on cherche à vérifier). Pour s'abonner à ce groupe multicast, il faut envoyer un message MLD Report (*Multicast Listener Discovery*) pour lequel l'adresse source sera l'adresse non spécifiée (`::`) et la destination l'adresse multicast `ff02::16` qui correspond à tous les routeurs du lien qui supportent le protocole MLD version 2.

Ensuite la station doit envoyer un message *neighbor solicitation* dans lequel l'adresse source sera l'adresse non spécifiée (`::`) et l'adresse destination sera l'adresse multicast `ff02::1:ff45:6789`.

Par défaut, une station ne transmet qu'un *neighbor solicitation*. Sans réception d'un *neighbor advertisement*, la station A en déduit que l'adresse `fe80::201:23ff:fe45:6789` est unique et peut donc l'utiliser.

Ensuite la station A cherche à créer une adresse globale. Pour ce faire elle envoie un *router solicitation* dont la source est l'adresse `fe80::201:23ff:fe45:6789` et la destination est l'adresse multicast `ff02::2` (qui correspond à tous les routeurs IPv6 du lien). Peu de temps après, elle devrait recevoir un message *router advertisement* dont l'adresse source correspond à l'adresse IPv6 lien-local du routeur

émetteur et l'adresse destination correspond à l'adresse multicast `ff02::1` (tous les noeuds IPv6 du lien).

Avec le préfixe reçu dans le *router advertisement*, la station peut créer une adresse globale : `2001:db8:beef::201:23ff:fe45:6789` et ensuite vérifier cette adresse via l'émission d'un *neighbor solicitation* dont l'adresse source est l'adresse non spécifiée (`::`) et l'adresse destination est l'adresse multicast `ff02::1:ff45:6789`. Sans réponse, la station A considère que cette adresse est unique et peut l'utiliser.

Au final, il y a 5 messages échangés pour la configuration de la station A. La station B se trouvant dans un autre réseau, il y aura également 5 messages échangés sur le réseau L3.

4. Calculez le temps maximal nécessaire pour réaliser l'autoconfiguration sans état en supposant que le support de communication est sans erreur.

La transmission / réception / traitement des messages est négligeable. Lorsqu'une interface a été initialisée, il faut attendre un délai entre 0 et `MAX_RTR_SOLICITATION_DELAY` (qui vaut 1 seconde par défaut) avant de transmettre le MLD report. Donc ici, dans le pire des cas, une station doit attendre 1 seconde. Après la réception d'un *router advertisement* envoyé en multicast, une station doit également attendre un délai similaire avant l'émission du *neighbor solicitation*, donc 1 seconde de plus d'attente. Après l'émission d'un *neighbor solicitation* (lors d'un DAD), une station doit attendre 1 seconde (`RETRANS_TIMER`). Lorsque ce temporisateur expire et que la station n'a pas reçu de *neighbor advertisement* alors elle en déduit que l'adresse est unique. Ensuite, les routeurs sont par défaut configurés pour ne pas répondre aux sollicitations et transmettre périodiquement des *router advertisements* toutes les `X` secondes, `X` étant compris entre 200 et 600 secondes. Dans le pire des cas, les stations devront donc attendre 600 secondes. Ce qui donne au final un temps de 604 secondes dans le pire des cas.

5. Quelles routes faut-il fournir explicitement aux stations ? Aux routeurs ?

Il ne faut fournir aucune route spécifique aux stations. Grâce à l'auto-configuration sans état elles ont toutes obtenues une route par défaut. Concernant les routeurs, il faut fournir une route pour joindre L3 à R1 et une route pour joindre L1 à R2.

6. La station A envoie un paquet IP à la station B. Ce paquet traverse les routeur R_1 et R_2 . On suppose que toutes les tables des voisins (*Neighbor Cache*) sont vides. Donnez la liste des trames transmises par l'envoi de ce paquet. Pour chaque trame précisez les adresses Ethernet, le protocole transporté et les adresses de niveau 3 si présentes.

A doit dans un premier temps résoudre l'adresse de niveau 2 de R_1 via l'émission d'un *neighbor solicitation* :

- adresse source Ethernet : @mac de A
- adresse destination Ethernet : `33:33:33` + 32 derniers bits de l'adresse destination IP (adresse multicast Ethernet pour IPv6)
- adresse source IP : @lien-local de A
- adresse destination IP : @multicast du noeud sollicité correspondant à l'adresse lien-local de R_1

Après réception, R_1 répond par un *neighbor advertisement* :

- adresse source Ethernet : @mac de R_1
- adresse destination Ethernet : @mac de A
- adresse source IP : @lien-local de R_1
- adresse destination IP : @lien-local de A

A peut maintenant envoyer son paquet IP contenant les données pour B :

- adresse source Ethernet : @mac de A
- adresse destination Ethernet : @mac de $R1$
- adresse source IP : @globale de A
- adresse destination IP : @lglobale de B

À la réception de ce paquet, $R1$ doit le transmettre à $R2$ via le réseau L_2 . Dans un premier temps il doit résoudre l'adresse de niveau 2 de $R2$ via l'émission d'un *neighbor solicitation* :

- adresse source Ethernet : @mac de $R1$
- adresse destination Ethernet : 33 : 33 + 32 derniers bits de l'adresse destination IP (adresse multicast Ethernet pour IPv6)
- adresse source IP : @lien-local de $R1$
- adresse destination IP : @multicast du noeud sollicité correspondant à l'adresse lien-local de $R2$

Après réception, $R2$ répond par un *neighbor advertisement* :

- adresse source Ethernet : @mac de $R2$
- adresse destination Ethernet : @mac de $R1$
- adresse source IP : @lien-local de $R2$
- adresse destination IP : @lien-local de $R1$

$R1$ peut maintenant transmettre le paquet IP à destination de B :

- adresse source Ethernet : @mac de $R1$
- adresse destination Ethernet : @mac de $R2$
- adresse source IP : @globale de A
- adresse destination IP : @lglobale de B

À la réception de ce paquet, $R2$ doit le transmettre à B via le réseau L_3 . Dans un premier temps il doit résoudre l'adresse de niveau 2 de B via l'émission d'un *neighbor solicitation* :

- adresse source Ethernet : @mac de $R2$
- adresse destination Ethernet : 33 : 33 + 32 derniers bits de l'adresse destination IP (adresse multicast Ethernet pour IPv6)
- adresse source IP : @globale de $R2$
- adresse destination IP : @multicast du noeud sollicité correspondant à l'adresse globale de B

Après réception, B répond par un *neighbor advertisement* :

- adresse source Ethernet : @mac de B
- adresse destination Ethernet : @mac de $R2$
- adresse source IP : @globale de B
- adresse destination IP : @globale de $R2$

$R2$ peut maintenant transmettre le paquet IP à destination de B :

- adresse source Ethernet : @mac de $R2$
- adresse destination Ethernet : @mac de B
- adresse source IP : @globale de A
- adresse destination IP : @lglobale de B

- On suppose que la station B tombe en panne. Est-ce que la station A est en mesure de détecter cette panne ? Si oui calculez le temps maximal nécessaire pour détecter cette panne. Sinon, proposez un mécanisme qui permette une telle détection.

A ne peut pas utiliser le mécanisme *neighbor unreachability detection* pour détecter la panne de B car les deux stations se situent dans des réseaux IP différents. La seule possibilité pour A est d'envoyer des paquets IP à B et de recevoir des messages *ICMP* de type 3 (*destination unreachable*) avec le code

1 (*destination host unreachable*) de la part de *R2*. *R2* doit être configuré pour envoyer des messages ICMP en cas d'erreurs.