TP Réseau n°13 - IPv6

Sommaire

*Un compte-rendu au format texte (**UTF-8**) devra être rédigé et envoyé à l'adresse*
tvaira@free.fr
La convention de nommage pour les compte-rendus est la suivante : ***tp-13-nom.txt***

Introduction

Mise en situation

Vous devez disposer d'un PC possédant un système d'exploitation Linux ou Windows et du logiciel de virtualisation *VirtualBox*. Le système invité sera l'ISO du Live CD Raizo fourni intégrant le logiciel de virtualisation **Netkit**.

Remarque : il est conseillé de consulter la FAQ Netkit en cas de besoin.

Les adresses IPv6

Les adresses IPv6 sont codées sur **128 bits** soit **16 octets**.

La notation décimale pointée employée pour les adresses IPv4 (par exemple `172.31.128.1`) est abandonnée au profit d'une écriture hexadécimale, où les 8 groupes de 2 octets (soit 16 bits par groupe) sont séparés par un signe deux-points ':' : `2001:0db8:0000:85a3:0000:0000:ac1f:8001`.

La notation complète comprend exactement 39 caractères.

Il est permis d'omettre de 1 à 3 chiffres zéros non significatifs dans chaque groupe de 4 chiffres hexadécimaux. Ainsi, l'adresse IPv6 ci-dessus est équivalente à : `2001:db8:0:85a3:0:0:ac1f:8001`.

De plus, une unique suite de un ou plusieurs groupes consécutifs de 16 bits tous nuls peut être omise, en conservant toutefois les signes deux-points de chaque côté de la suite de chiffres omise, c'est-à-dire une paire de deux-points (::). Ainsi, l'adresse IPv6 ci-dessus peut être abrégée en : `2001:db8:0:85a3::ac1f:8001`.

Pour les cas où le ':' a un sens (par exemple dans une URL), on met l'adresse IPv6 entre `[]` pour éviter toute confusion. Exemple : `http://[::1]/`

La notion historique de classes a totalement disparu, au profit de l'utilisation exclusive des préfixes et de la notation CIDR avec le slash et le masque, déjà utilisés en IPv4. Les masques par défaut disparaissent donc.

Les définitions des adresses IP version 6 sont documentées dans les RFC (*Request For Comments*) suivants :
– RFC 2460 - *Internet Protocol, Version 6 (IPv6) Specification*, décembre 1998
– RFC 2373 - *IP Version 6 Addressing Architecture*, juillet 1998
– RFC 2893 - *Transition Mechanisms for IPv6 Hosts and Routers*, août 2000

Question 1. Donner une écriture en forme abrégée pour les adresses suivantes :

a) `2001:0001:0002:014E:F140:0102:8012:00AE`

b) `FEDC:0000:0000:0000:0400:A987:6543:210F`

c) `1FFF:0000:0A88:85A3:0000:0000:0C10:8001`

d) `FE80:0000:0000:0000:0000:0000:0000:0001`

Question 2. Est-ce que les adresses suivantes sont des adresses IPv6 valides ?

a) `2001:14C8::871:206:A14:23`

b) `2001:14C8::871:206::A14:23`

c) 2001:14C8:0:0134::A120:E001

d) 200F:23G5:23:1:45:A234::1

Question 3. Écrire sous la forme complète les adresses IPv6 suivantes :

a) 2001:14C8::871:206:A14:23

b) 2002:203::AEF:12:0:1B1:1

c) 2003::2

d) 2001::45:0:6

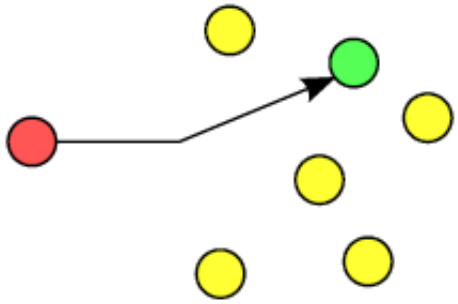
Question 4. Déterminer le numéro de réseau de l'adresse 2001:0660:2402:1001:208:2ff:fedc:6133/48.

Question 5. Déterminer le numéro de réseau de l'adresse et l'identifiant d'interface de l'adresse 2001:0660:F402:1000:208:2ff:fedc:9033/64.

Manipulations

Séquence 1 : adresse unicast

Il s'agit du même type d'adresse (RFC 4291, section 2.5) qu'en IPv4. C'est donc le type d'adresse le plus classique, composé d'un préfixe réseau à longueur variable suivi d'un identifiant d'interface (parfois dérivé des adresses MAC sur 64 bits). C'est une adresse utilisable comme adresse source ou comme destination.



On distingue trois types d'adresses unicast :

- **Global Unicast** : adresse globale. Il s'agit des adresses qui sont uniques dans le monde, et qui sont par conséquent routables sur Internet. Elles se composent d'un préfixe de routage global (/48), suivi du préfixe de sous-réseau (/16) et de l'identifiant d'interface (/64). L'IANA fournit une liste des préfixes affectés aux différents RIR qui permettent donc de classer les IP rencontrées sur le réseau mondial par région du monde. Le préfixe 2001:db8::/32 est réservé pour les exemples des documentations (RFC 3849) et il n'est donc pas routable sur Internet.
- **Link-local Unicast** : adresses de lien local, non-routables en local comme sur Internet. Elles utilisent toutes le préfixe fe80::/10. Elles sont systématiquement générées lors de l'utilisation de l'autoconfiguration sans état (*stateless*).
- **Unique Local Unicast** : adresses de lien local, non-routables sur Internet. C'est la catégorie qui se rapproche le plus des adresses privées IPv4 (RFC 1918). Elles utilisent toutes le préfixe fc00::/7, mais avec le huitième bit en partant de la gauche positionné à 1 si le préfixe est défini localement. Puisque la valeur 0 n'est pas possible actuellement (usage futur), elles sont reconnaissables par leur premier bloc qui commence systématiquement par fd.

Structure des adresses unicast globales

champ	*préfixe*	*sous-réseau*	*interface*
bits	48	16	64

Structure des adresses link-local

champ	*préfixe*	*zéro*	*interface*
bits	10	54	64

1111111010

Structure des adresses locale unique

champ	*préfixe*	*L*	*ID globale*	*Subnet*	*Interface*
bits	7	1	40	16	64

1111110

Concernant l'**adresse de boucle locale** (*loopback*), les administrateurs qui ne voient pas beaucoup le soleil et qui ont `127.0.0.1` écrit sur leur paillasson, devront le troquer pour une version plus récente avec `::1` inscrit dessus. L'adresse :: correspond logiquement à l'adresse IPv4 `0.0.0.0` et ne sera donc utilisée que pour définir les passerelles par défaut, ou comme adresse source des paquets de découverte de son IP.

La construction automatique de l'adresse IP lors de l'activation d'une interface suit le principe suivant :
- Ajouter les octets `fffe` au milieu de l'adresse MAC de l'interface.
- Positionner le septième bit de l'adresse MAC modifiée en partant de la gauche à 1 si l'adresse est unique (ce qui est le cas pour toutes les adresses MAC par défaut) sinon 0.
- Récupération du préfixe si c'est une adresse globale, sinon **utilisation du préfixe d'adresse de lien** local.
- Concaténation du préfixe avec l'adresse MAC ainsi modifiée.

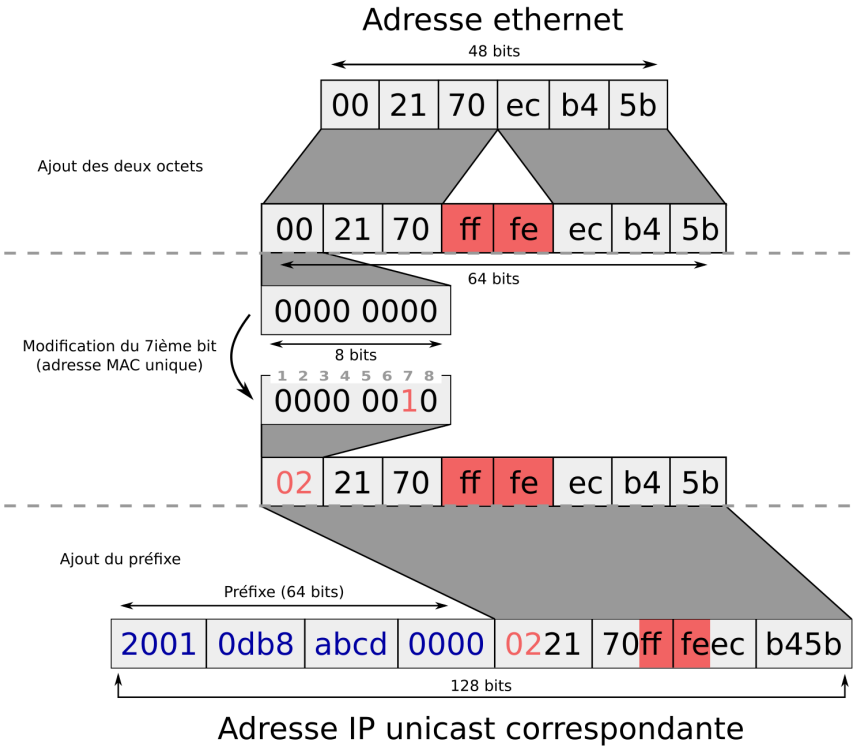


FIGURE 1 – *EUI-64 formé depuis l'EUI-48 correspondant à l'adresse MAC*

Pour cette séquence, la configuration de la maquette est la suivante (archives `tp2.tgz` ou `tp13.tgz`) :

Question 6. Relever la configuration des interfaces du poste **pc1**.Quelle est l'adresse IPv6 donnée à l'interface `lo`.

```
pc1:~# ifconfig
pc1:~# ip -6 addr ls dev lo
```

Question 7. Donner la commande qui permet d'envoyer 3 paquets `ICMP` *echo request* vers l'interface `lo` en IPv6. Quel est le nouveau nom de la commande `ping` pour IPv6 ?

```
pc1:~# ping6 -c 3 ::1
```

Question 8. À partir du fichier `/etc/hosts`, donner le(s) nom(s) associé(s) à l'adresse de boucle locale en IPv6 ?

```
pc1:~# cat /etc/hosts
```

Question 9. Activer l'interface `eth0` du poste **pc1** et relever sa configuration.

```
pc1:~# ifconfig eth0 up
pc1:~# ifconfig
pc1:~# ip -6 addr ls dev eth0
```

Question 10. Déterminer le type d'adresse affectée à l'interface `eth0`.

```
pc1:~# ipv6calc -qi fe80::6c5f:98ff:fe37:c07
```

Question 11. Quelle serait l'adresse « *link local* » d'une machine dont l'adresse MAC est `00:4f:4e:08:25:1a` ?

Question 12. On va maintenant ajouter une adresse IPv6 à l'interface `eth0` du poste `pc1`. Vérifier avec la commande `ping6`.

```
pc1:~# ifconfig eth0 inet6 add 2001:db8:46::1/64
pc1:~# ifconfig
```

Une interface peut avoir plusieurs adresses IPv6 en même temps. En fait, une interface aura au moins 2 adresses : une adresse globale et une adresse link-local qui est allouée à l'activation de l'interface et sert aux protocoles d'initialisation. Plusieurs préfixes `/64` peuvent donc être routés sur le même lien.

Question 13. Quel est le type de cette adresse ?

```
pc1:~# ipv6calc -qi 2001:db8:46::1
pc1:~# ifconfig | grep -i scope
```

Question 14. Afficher la table de routage du poste `pc1`.

```
pc1:~# route -n -A inet6
```

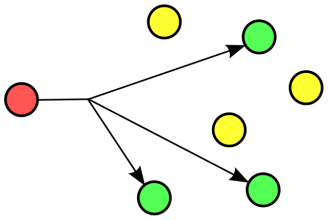
Question 15. Démarrer le service `ssh` puis observer les connexions réseaux pour la pile IPv6.

```
pc1:~# /etc/init.d/ssh start
pc1:~# netstat -natup -A inet6
```

Question 16. Les adresses *unicasts* peuvent être classées en deux catégories d'usages. Lesquelles ?

Séquence 2 : adresse multicast

Ce type d'adresse (RFC 4291, section 2.7) correspond précisément à son homologue IPv4. Il s'agit d'adresses virtuelles qui distribuent des paquets à tous les membres inscrits dans un groupe. C'est une adresse utilisable comme adresse destination.



Elles utilisent toutes le préfixe `ff00::/8`. Elles sont donc reconnaissables par leur premier bloc qui commence systématiquement par `ff`.

Format d'une adresse multicast

champ	*préfixe*	*drap.*	*scope*	*groupe*
bits	8	4	4	112

11111111

Les quatre bits qui suivent le préfixe désignent les drapeaux (*flags* ORPT). Les quatre bits qui suivent les drapeaux correspondent à la portée de l'adresse (*scope*), qui peuvent limiter son rayonnement à l'interface (1) pour les tests, au lien local (2), au site correspondant au réseau local (5), à Internet (e) ou à des zones plus variées décrites dans la RFC 4291.

Un certain nombre d'adresses multicast sont normalisées par l'IETF (drapeau T positionné à 0), et sont documentées dans la RFC 2461.

Nom	Adresse	Équivalent IPv4	Fonction
all-nodes	`ff02::1`	`224.0.0.1`	Tous les noeuds et routeurs du lien local (utilisée par exemple pour les interfaces qui n'ont pas encore d'adresse mais qui veulent recevoir une réponse)
all-routers	`ff02::2`	`224.0.0.2`	Tous les routeurs du lien local (utilisée par exemple pour solliciter une annonce de préfixe sur le réseau)

Well-known IPv6 multicast addresses

Address	Description
ff02::1	All nodes on the local network segment
ff02::2	All routers on the local network segment
ff02::5	OSPFv3 All SPF routers
ff02::6	OSPFv3 All DR routers
ff02::8	IS-IS for IPv6 routers
ff02::9	RIP routers
ff02::a	EIGRP routers
ff02::d	PIM routers
ff02::16	MLDv2 reports (defined in RFC 3810)
ff02::1:2	All DHCP servers and relay agents on the local network segment (defined in RFC 3315)
ff02::1:3	All LLMNR hosts on the local network segment (defined in RFC 4795)
ff05::1:3	All DHCP servers on the local network site (defined in RFC 3315)
ff0x::c	Simple Service Discovery Protocol
ff0x::fb	Multicast DNS
ff0x::101	Network Time Protocol
ff0x::108	Network Information Service
ff0x::114	Used for experiments

Le *broadcast* a été supprimé et ne doit plus être pris en considération dans les plans d'adressage, au profit de l'utilisation massive du multicast. Le groupe multicast qui se rapproche le plus du fonctionnement du *broadcast* traditionnel correspond au groupe *all-nodes* donné en exemple et censé correspondre à tous les noeuds (mais dont l'inscription est à la discrétion de la machine).

L'adresse ethernet (MAC) d'une trame à destination d'un groupe multicast est dérivée de l'adresse IP du groupe (RFC 5342 et 2464) :

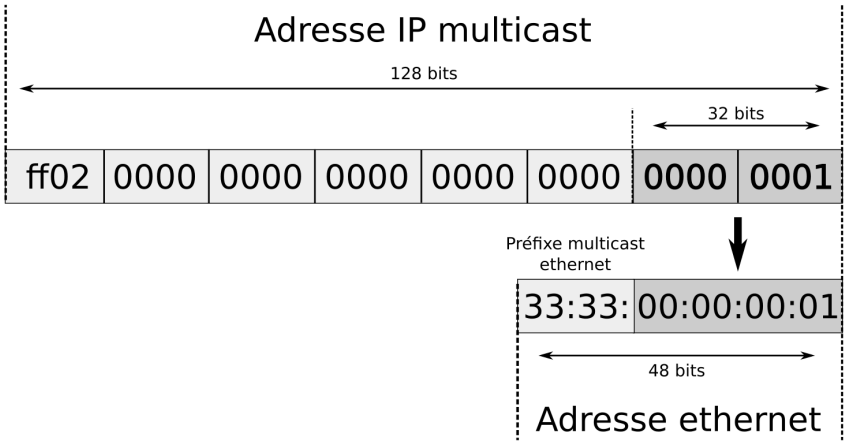


FIGURE *2 – Conversion d'une adresse IP multicast en adresse ethernet multicast (adresse all-nodes)*

Question 17. Afficher le contenu du fichier `/etc/hosts`. Contient-il des adresses multicast ? Si oui lesquelles ?

Question 18. Afficher les abonnements multicast pour le poste `pc1` ?

```
pc1:~# nestat -g6n
pc1:~# ip -6 maddress show dev eth0
```

Séquence 3 : protocole IPv6

Question 19. On va maintenant ajouter une adresse IPv6 à l'interface `eth0` du poste `r1`. Vérifier avec la commande `ping6`.

```
r1:~# ifconfig eth0 inet6 add 2001:db8:46::2/64
r1:~# ifconfig
```

Question 20. Emettre un message ICMP de `r1` vers `pc1`. Donner l'option de la commande `ping6` qui permet d'envoyer un seul message ICMP.

Activer une capture `wireshark` sur le domaine A.

Question 21. En vous aidant de la capture réalisée, répondre aux questions suivantes :

a) Quelle est la valeur du champ `Type` des trames Ethernet_II échangées ? À quoi sert ce champ dans une trame Ethernet_II ?

b) Quelle est la valeur du champ `Version` des paquets IP échangés ?

c) Quelle est la valeur du champ `Next header` des paquets IP échangés ? À quoi sert ce champ dans le protocole IPv6 ? Vérifier sa valeur en consultant le fichier `/etc/protocols`.

d) Quelle est la valeur du champ `Payload length` des paquets IP échangés ? À quoi sert ce champ dans le protocole IPv6 ?

e) Quel est le **type** et le **code** des messages ICMP envoyés et reçus dans cet échange ?

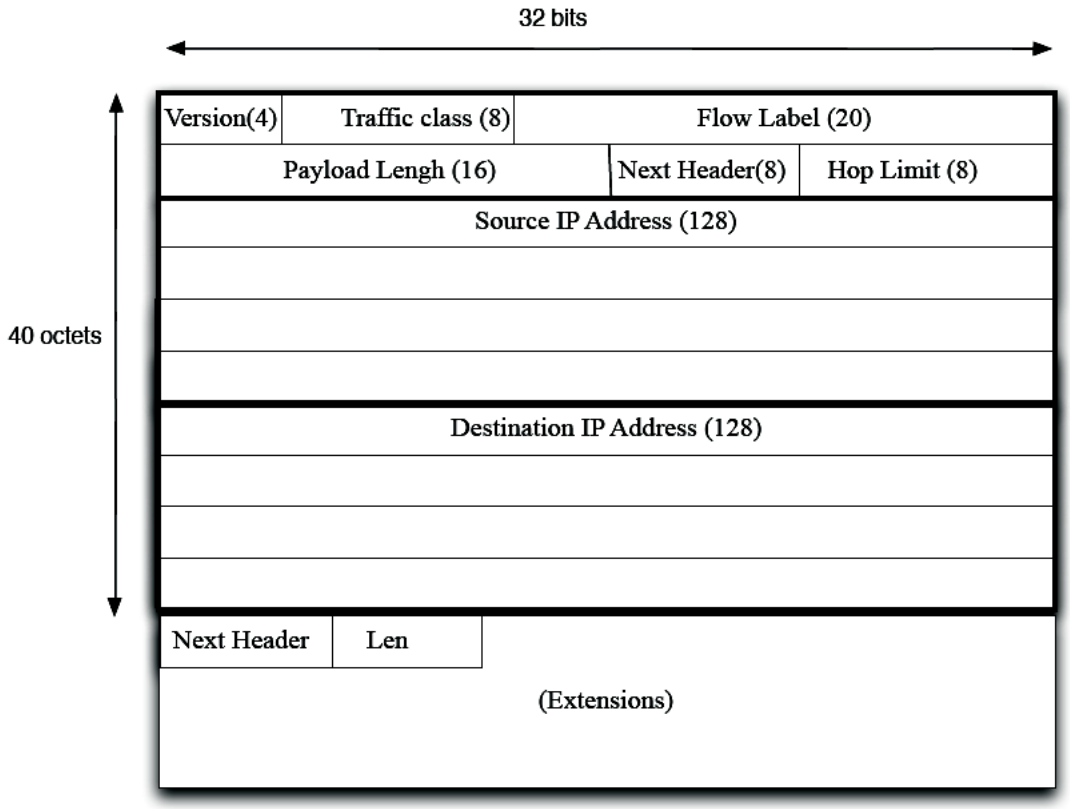


FIGURE *3 – Le protocole IPv6*

Question 22. En vous aidant de la capture réalisée, répondre aux questions suivantes :

a) Observez-vous des trames ARP et des adresses de diffusion générale (*broadcast*) ?

b) Qu'indique la valeur utilisée comme adresse destination dans la trame Ethernet_II ?

c) Quel est le rôle des paquets « *icmpv6 neighbor solicitation* » ?

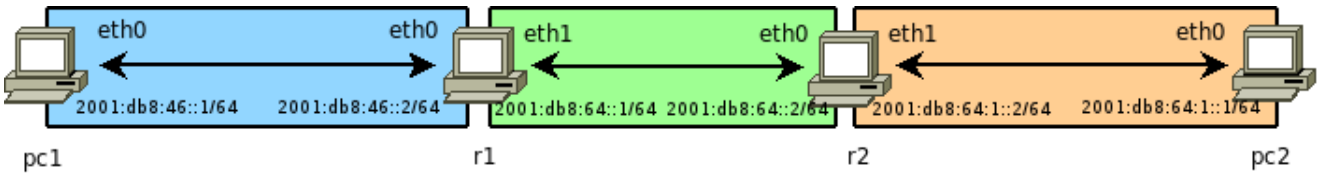
Question 23. Consulter le cache des voisins sur les machines `pc1` et `r1` avec la commande ci-dessous, tout comme on le faisait sous IPv4 avec la commande `arp`.

```
pc1:~# ip neigh show
```

Après quelques secondes, on observera qu'il y a un temps d'expiration pour les entrées et la table de voisinage sera vidée.

Séquence 4 : routage statique IPv6

Soit le plan d'adressage IPv6 suivant (archives `tp2.tgz` ou `tp13.tgz`) :



Les adresses constituées entièrement de 0 ou de 1 ne jouent pas de rôle particulier en IPv6. En IPv6, les sous-réseaux ont une taille fixe de /64, c'est-à-dire que 64 des 128 bits de l'adresse IPv6 sont réservés à la numérotation d'un hôte dans le sous-réseau. En IPv6, les masques de sous-réseaux ont donc une taille fixe de /64.

Rappels :

```
// Activer l'interface eth0
pc1:~# ifconfig eth0 up

// Affecter une adresse IPv6 à l'interface eth0
pc1:~# ifconfig eth0 inet6 add 2001:db8:46::1/64

// Afficher la table de routage de la pile IPv6
pc1:~# route -n -A inet6
// ou :
pc1:~# ip -6 route list dev eth0
```

Pour ajouter une route indirecte à la table de routage d'une machine Linux, il faudra faire par exemple :

```
pc1:~# route -A inet6 add 2001:db8:64::/64 gw 2001:db8:46::2 dev eth0
// ou :
pc1:~# ip -6 route add 2001:db8:64::/64 via 2001:db8:46::2 dev eth0
```

Pour faire office de routeur, les postes Linux `r1` et `r2` doivent relayer (retransmettre) les paquets sur les réseaux qu'ils interconnectent. Pour cela, il faut activer le *forwarding* :

```
r1:~# echo "1" > /proc/sys/net/ipv6/conf/default/forwarding
r1:~# echo "1" > /proc/sys/net/ipv6/conf/all/forwarding
```

Question 24. Configurer les tables de routage de toutes les machines afin que chaque PC puisse communiquer avec n'importe quel autre. Fournir les tables de routages des 4 postes. Tester avec `ping6`.

On n'utilisera pas de routes par défaut pour ces machines.

Séquence 5 : auto-configuration avec radvd

Les interfaces peuvent être configurées manuellement ou s'auto-configurer.

Dans le cas de l'auto-configuration sans état (*stateless*) [RFC2462], où seul le préfixe est donné, l'équipement aura la charge de générer le suffixe de l'adresse. Dès qu'une interface est activée (démarrage de la machine, par exemple), une adresse de type lien local est automatiquement générée à partir de l'adresse MAC de l'interface.

Par exemple, une carte réseau d'adresse MAC 00 :0D :61 :22 :34 :76 aura l'adresse IPv6 fe80 : :20d :61ff :fe22 :3476.

Le démon `radvd` (*Router ADVertisement Daemon*) permet d'auto-configurer toutes les interfaces du réseau avec un autre préfixe de réseau (par exemple, celui qui vous aura été fourni par votre FAI, celui que vous aurez choisi pour vos tests ou encore votre préfixe ULA) en utilisant un routeur IPv6 sur lequel `radvd` est installé. Cela doit permettre de configurer les adresses des noeuds de type lien global (Scope Global).

Le démon `radvd` doit être installé sur un des postes de votre réseau (ici `r1`). Il est utilisé sur les systèmes GNU/Linux pour communiquer des informations aux clients quand ils doivent être auto-configurés. `radvd` envoie régulièrement sur un réseau local Ethernet des messages classés **RA** (*Router Advertisement*). Il est également habilité à répondre à des requêtes de type **RS** (*Router Solicitation*). Ce processus fait partie du protocole **ND** (*Neighbor Discovery*), ou protocole de découverte des voisins [RFC2461].

Lire : http ://fr.wikipedia.org/wiki/Neighbor_Discovery_Protocol

Pour rappel, la construction automatique de l'adresse IP suit le principe suivant :
- Ajouter les octets `fffe` au milieu de l'adresse MAC de l'interface.
- Positionner le septième bit de l'adresse MAC modifiée en partant de la gauche à 1 si l'adresse est unique (ce qui est le cas pour toutes les adresses MAC par défaut) sinon 0.
- **Récupération du préfixe si c'est une adresse globale**, sinon utilisation du préfixe d'adresse de lien local.
- Concaténation du préfixe avec l'adresse MAC ainsi modifiée.

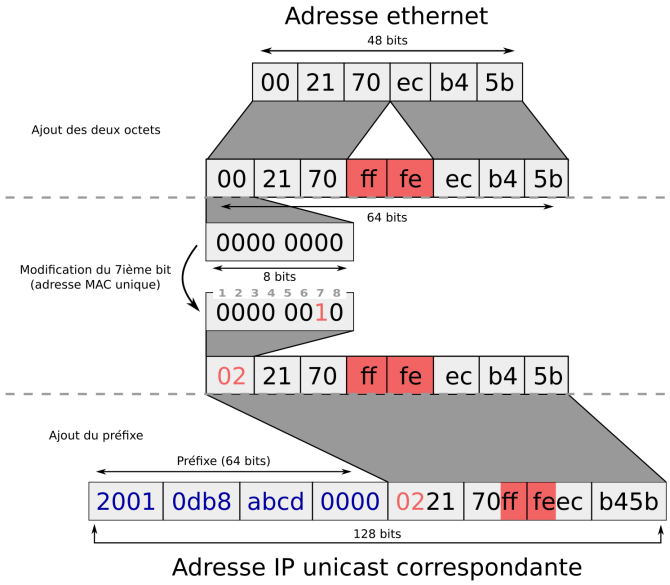
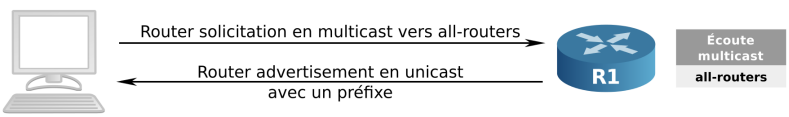
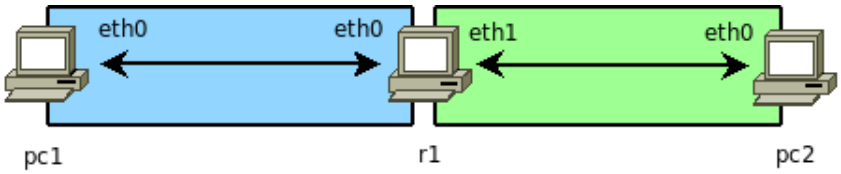


FIGURE 4 – *EUI-64 formé depuis l'EUI-48 correspondant à l'adresse MAC*

Le préfixe global s'obtient en écoutant les diffusions **ICMPv6** des *Router Advertisement* sur le réseau. En général ce sont les routeurs qui se chargent de diffuser les préfixes périodiquement, mais une machine a la possibilité de réclamer cette diffusion immédiatement en utilisant un message **ICMPv6** de type *Router Solicitation* à destination de l'adresse *all-routers*.



La configuration du réseau est la suivante :



Pour cette séquence, vous devez utiliser la maquette fournie dans l'archive `tp13-radvd.tar.gz`.

Le démarrage du routeur `r1` exécute les commandes suivantes :

```
# Active le forwarding (relayer les paquets IPv6)
echo 1 > /proc/sys/net/ipv6/conf/all/forwarding
# Active l'interface eth0
/sbin/ifconfig eth0 up
# Active l'interface eth1
/sbin/ifconfig eth1 up
# Affecte une adresse IPv6 à l'interface eth0
/sbin/ifconfig eth0 inet6 add 2001:db8:46:0::ff/64
/bin/chmod 664 /etc/radvd.conf
# Installe le démon radvd sur la machine
/usr/bin/dpkg -i /opt/radvd-1.6-1_i386.deb
# Démarre le démon radvd
/etc/init.d/radvd restart
```

Question 25. Consulter le fichier de configuration `/etc/radvd.conf`. Quel est le préfixe diffusé par le routeur `r1` sur son interface `eth0` ?

Lire : http ://linux.die.net/man/5/radvd.conf

Activer une capture `wireshark` sur le domaine ip46.

Activer l'interface `eth0` de `pc1` :

```
pc1:~# ifconfig eth0 up
pc1:~# ifconfig eth0
```

Question 26. En vous aidant de la capture réalisée, observer le message **RS** et répondre aux questions suivantes :

a) Quel est le type de message ICMPv6 envoyé par la machine `pc1` ?

b) Quelle est l'adresse source du paquet contenant ce message ? Quelle est son type ?

c) Quelle est l'adresse destination du paquet contenant ce message ? Que signifie-t-elle ?

Question 27. En vous aidant de la capture réalisée, observer le message **RA** et répondre aux questions suivantes :

a) Quel est le type de message ICMPv6 reçu par la machine `pc1` ?

b) Quelle est l'adresse source du paquet contenant ce message ? Quelle est son type ?

c) Quelle est l'adresse destination du paquet contenant ce message ? Que signifie-t-elle ?

d) Quelle est l'information envoyée par `r1` dans le message ICMPv6 ? Quelle est sa durée de vie ?

Question 28. Afficher la table de routage de la machine `pc1`.

```
pc1:~# route -n -A inet6
pc1:~# ip -6 route list dev eth0
```

Question 29. En vous aidant de l'affichage de la table de routage de `pc1`, répondre aux questions suivantes :

a) Possède-t-elle une route par défaut ?

b) Si oui, comment l'a-t-elle obtenue ?

c) Quelle est alors l'adresse de la passerelle par défaut ?

c) Quelle est la durée de vie de cette route (cf. message RA précédent) ?

Question 30. Modifier la configuration du routeur `r1` (fichier `radvd.conf`) pour qu'il diffuse le préfixe `2001:db8:64::/64` sur son interface `eth1`.

Question 31. Vérifier que la machine `pc2` a obtenu une adresse IPv6. Pinger la machine `pc1`.

Attention : n'oubliez pas d'affecter une adresse IPv6 à l'interface `eth1` du routeur `r1` dans le préfixe `2001:db8:64::/64`.

Séquence 6 : auto-configuration stateful (DHCPv6)

Cette méthode d'auto-configuration utilise un serveur DHCPv6, de façon très similaire au protocole DHCPv4, moyennant un léger reconditionnement des messages échangés et l'utilisation des adresses *anycast* (RFC 3315).

Pour recenser la liste des serveurs DHCP du réseau, le client enverra un message de type `SOLICIT` (correspondant à l'ancien `DHCPDISCOVER`) à l'adresse *anycast* du routeur du sous-réseau qui renverra un message de type `ADVERTISE` (anciennement `DHCPOFFER`).

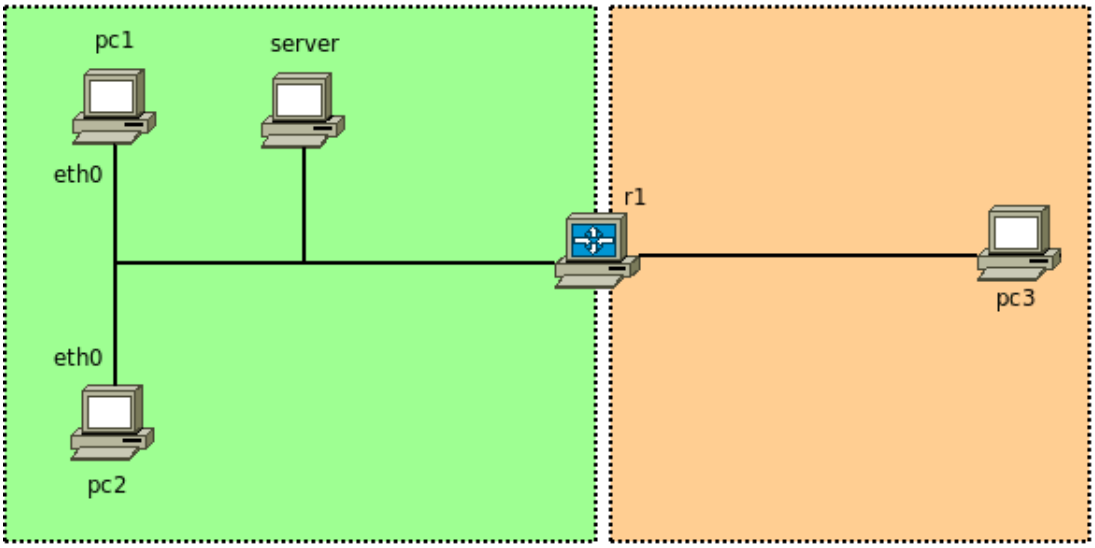
Les adresses *anycast* (RFC 4291, section 2.6) ne sont pas différenciables des adresses *unicast* (seule les interfaces qui les utilisent savent qu'elles sont *anycast*). Elles permettent de contacter une machine parmi un lot sans en avoir conscience. Si les machines qui le composent sont à des distances différentes, ce seront les routeurs intermédiaires qui choisiront de router les paquets à celles qui sont les plus proches topologiquement (métrique la plus basse). Si plusieurs machines du lot sont à la même distance, elles répondront toutes, et seule la réponse la plus rapide sera prise en compte par l'expéditeur initial. Ce type d'adresse peut donc être affecté à plusieurs interfaces sur un réseau.

Les messages DHCP (envoyés en ICMPv6) qui existaient dans DHCPv4 changent de nom (l'ancien nom est rappelé entre parenthèses), et des nouveautés sont ajoutées :
- `SOLICIT` (`DHCPDISCOVER`) : Permet au client de faire un appel général sur son réseau pour localiser les serveurs DHCPv6. Le message est envoyé en multicast à l'adresse `ff02::1:2` (RFC 3315).
- `ADVERTISE` (`DHCPOFFER`) : Réponse des serveurs DHCPv6 aux `SOLICIT`, en unicast.
- `REQUEST` (`DHCPREQUEST`) : Utilisé par le client pour demander les paramètres de configuration au serveur DHCPv6 sélectionné.
- `CONFIRM` : C'est une nouveauté, et elle permet au client de simplement s'assurer auprès du serveur que son ou ses adresses sont toujours valides.
- `RENEW` (`DHCPREQUEST`) : Permet au client de demander un prolongement du bail, ou mettre à jour ses paramètres si ceux-ci ont changés depuis.

- `REBIND` (`DHCPREQUEST`) : Ce message a la même fonctionnalité que le `RENEW`, mais il est utilisé pour interroger en multicast l'ensemble des serveurs DHCPv6, lorsque le serveur DHCP d'origine ne répond plus.
- `RELEASE` (`DHCPRELEASE`) : Envoyé par le client au serveur pour indiquer à ce dernier la valeur des paramètres actuellement utilisés.
- `DECLINE` (`DHCPDECLINE`) : Permet au client de signifier au serveur que les paramètres transmis ne peuvent pas être utilisés.
- `REPLY` (`DHCPACK` et `DHCPNACK`) : Il s'agit du message contenant la réponse du serveur DHCP pour les deux types d'interrogations `RENEW` et `REBIND`, ou pour confirmer la bonne réception des deux types de messages précédents.
- `INFORMATION-REQUEST` (`DHCPINFORM`) : Permet simplement au client de demander de nouveaux paramètres de configuration.
- `RECONFIGURE` (`DHCPFORCERENEW`) : Permet au serveur d'indiquer à ses clients que les paramètres doivent être actualisés, et qu'il serait bon qu'ils le sollicitent avec un message `RENEW` ou `INFORMATION-REQUEST`.
- `RELAY-FORWARD` : C'est une seconde nouveauté, et c'est utilisé par les relais pour transmettre à un serveur (ou un autre relais) le message initial du client, qui sera contenu dans les options.
- `RELAY-REPLY` : Il s'agit du pendant du type précédent, permettant de répondre à la question du client initial au travers d'un relais.

Comme dans sa version 4, le DHCP pourra retourner un domaine ainsi que des adresses DNS. La mise en place de relais est toujours possible.

La configuration du réseau est la suivante :



Pour cette séquence, vous devez utiliser la maquette fournie dans l'archive `tp13-dhcpd6.tar.gz`.

Le démarrage du serveur DHCPv6 `server` exécute les commandes suivantes :

```
# Active l'interface eth0
/sbin/ifconfig eth0 up
# Affecte une adresse IPv6 à l'interface eth0
/sbin/ifconfig eth0 inet6 add 2001:db8:46:0::2/64
```

Le démarrage du routeur `r1` exécute les mêmes commandes que lors de la séquence précédente. Mais son fichier de configuration `radvd.conf` est différent car il ne diffuse plus le préfixe mais seulement la **route par défaut** (car le protocole DHCPv6 ne permet plus d'indiquer la passerelle par défaut) pour le réseau relié à son interface `eth0` :

```
# cat /etc/radvd.conf

interface eth0 {
        AdvManagedFlag on;
        AdvOtherConfigFlag on;
        AdvSendAdvert on;
        prefix 2001:db8:46::/64 {
        AdvAutonomous off;
        };
};
interface eth1 {
        AdvSendAdvert on;
        prefix 2001:db8:64::/64 {
        AdvRouterAddr on;
        };
};
```

Question 32. Consulter le fichier de configuration `/etc/dhcp4/dhcpd.conf`. Quelle est la plage d'adresse dynamique utilisée par le serveur DHCPv6 ?

Activer une capture `wireshark` sur le domaine ip46.

Il faut démarrer tout d'abord le **service DHCPv6** sur la machine `server` :

```
server:~# /hostlab/dhcpd -6 -d -cf /etc/dhcp4/dhcpd.conf -lf /etc/dhcp4/dhcpd6.leases
```

Maintenant sur chaque machine cliente :

```
pc1:~# ifconfig eth0 up
pc1:~# touch ./dhcpd6.leases
pc1:~# /hostlab/dhclient -6 -lf ./dhcpd6.leases -sf /hostlab/linux
```

Vous devez observer des messages du service DHCPv6 sur la machine `server` :

```
Listening on Socket/5/eth0/2001:db8:46::/64
Sending on  Socket/5/eth0/2001:db8:46::/64
Solicit message from fe80::6c5f:98ff:fe37:c07 port 546, transaction ID 0x596CA200
Picking pool address 2001:db8:46::20
Sending Advertise to fe80::6c5f:98ff:fe37:c07 port 546
Request message from fe80::6c5f:98ff:fe37:c07 port 546, transaction ID 0x7C811B00
Sending Reply to fe80::6c5f:98ff:fe37:c07 port 546
...
```

On observe que l'adresse 2001:db8:46::20 a été proposée à une machine cliente.

Question 33. Afficher la configuration des interfaces et la table de routage de chaque machine cliente.

```
pc1:~# ifconfig
pc1:~# ip -6 addr ls dev eth0
pc1:~# route -n -A inet6
pc1:~# ip -6 route list dev eth0
```

Question 34. En vous aidant de la capture réalisée, observer les message **DHCPv6** et répondre aux questions suivantes :

a) Quel est le protocole de transport utilisé en DHCPv6 ?

b) Quels sont les numéros de port source et destination ?

c) Quelle est l'adresse destination des paquets contenant les messages `SOLICIT` et `REQUEST` ? Que signifie-t-elle ?

d) Représenter l'échange des messages DHCPv6 entre le client et le serveur ?

e) Quelle est l'adresse IPv6 allouée par le serveur ? Quelle est sa durée de vie ?

f) Quels sont les adresses des serveurs DNS et le domaine retournés par le serveur DHCPv6 ?

Question 35. Réaliser les tests de communication de `pc3` vers `pc1` puis de `pc3` vers `pc2`.

