# Introduction.

L'IPv4 utilise une représentation des adresses en 32 bits sous une forme décimale pointée de 4 chiffres décimaux (représentant chaque groupe de 8 bits), pouvant aller de 0 à 255 et séparés l'un de l'autre par un point.

En théorie, cette représentation permettait 2^32 adresses c'est à dire 4294967296. En tenant compte du fait que le nombre d'hôtes sur internet ayant besoin d'une adresse IP double chaque année, il fallait s'attendre à ce que le nombre d'adresses disponibles diminue rapidement. En février 2011, IANA distribuait ses 5 derniers blocs d'adresses disponibles. Les adresses restantes disponibles sont donc du ressort des différents RIR. L'utilisation des adresses en format CCIDR, l'affectation de certaines adresses réservées, le NAT, la demande auprès de client ayant reçu à une époque révolue des adresses en /8 de libérer les adresses non utilisées a permis de maintenir l'utilisation jusqu'à présent d'IPv4. Malgré ces démarches, il était nécessaire de définir un nouveau format permettant de se libérer des limites d'IPv4 dont la quantité limitée et permettre également un routage plus efficace.

La RFC 2373 (IP Version 6 Addressing Architecture) définit l'architecture des adresses IPv6.  
Une adresse IPv6 comporte 128 bits, c'est à dire 16 octets. S'il était aisé de représenter une adresse IPv4, en IPv6 il n'en est pas de même.

# Représentation d'une adresse IPv6.

Pour éviter d'avoir un nombre décimal énorme, les adresses IPv6 s'écrivent en hexadécimal avec ":" entre chaque bloc de deux octets. Il y a donc 8 blocs de 2 octets:

**abcd:ef11:2222:3333:4444:5555:6666:7777**

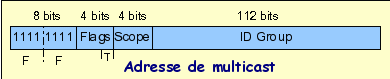
Chaque "chiffre" hexadécimal représente donc 4 bits, et est appelé un "nibble".

Dans l'écriture des adresses, chaque bloc est considéré séparément. Dans ce cadre, et comme en décimal ou en hexadécimal, les 0 non significatifs situés à gauche peuvent être éliminés:  
**0:1:2:a:4:5:b:7 -> 0000:0001:0002:000a:0004:0005:000b:0007**

Si plusieurs blocs consécutifs sont entièrement composés de 0, ils peuvent être non écrits grâce à une écriture particulière, par exemple:  
**12ab:0:0:0:0:0:0:1** -> **12ab::1**ou encore **1234:5678:0:0:abdc:0:0:ef01 -> 1234:5678::abdc:0:0:ef01**Dans ce dernier exemple, on voit qu'il ne faut pas utiliser plusieurs fois le " :: ". Un seul groupe de blocs consécutifs à 0 peut être simplifié dans une adresse.

# IPv6 et ses types d'adresses.

* Unicast: Cette adresse identifie une seule interface. Un paquet envoyé à une adresse Unicast est reçu par l'interface dépositaire de cette adresse.
* Multicast: Une telle adresse identifie un groupe défini de plusieurs interfaces de plusieurs nœuds différents. Un paquet émis vers une adresse multicast est reçu par tous les membres du groupe multicast. Les adresses multicast commencent par le préfixe **ff00::/8.**



Le bit T permet de définir la permanence de l'adresse. T=0, adresse permanente, T=1 adresse temporaire.

Le scope permet de définir la portée de la diffusion.

0 Réservé, 1 nœud, 2 lien, 3 sous réseau, 5 site, 8 organisation, E global, F Réservé

Certaines adresses étant prédéfinies, elles seront considérées comme permanentes. Voici quelques exemples... **ff02::1** toutes les machines du lien, **ff02::2**Tous les routeurs du lien, **ff05::2** tous les routeurs du site, **ff02::1:FFxx:xxxx** pour tous les noeuds se terminant par xx:xxxx dans leur adresse IP

* Anycast: Ce type d'adresse est nouveau par rapport à IPv4. Une adresse anycast identifie un groupe d'interfaces de plusieurs noeuds (comme en multicast). En revanche, lorsqu'un paquet est émis vers une adresse Anycast, il est reçu par l'un des membre du groupe anycast (n'importe lequel, mais a priori le plus proche en terme de routage réseau).

Nous avons évoqué dans le cours de réseau l'aspect pénalisant du broadcast sous IPv4. Sous IPv6, ce type d'adresse n'existe plus.

IPv6 admet des adresses particulières:

* Adresse de bouclage

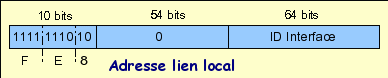
**::1**

* Adresse inconnue composée de tout des zéros

**::**

* Adresse de lien local (Scope Local)

Ces adresses sont attribuées aux cartes d'interface de façon dynamique. Elles sont valides uniquement dans un espace local ne traversant pas de routeur. Le préfixe d'une adresse de lien local est fe80::/10. Voici la structure d'une telle adresse:



Nous verrons plus tard dans les chapitres, la façon dont l'ID Interface peut être généré. Sous les systèmes d'exploitation Windows, la commande ipconfig fera apparaître les différentes adresses de lien local, une par interface. Cette IP de lien est indépendante de toute configuration. Elle sert notamment à adhérer aux divers groupes multicasts sur ce lien, qui remplacent totalement le broadcast ipv4 et l'ARP.

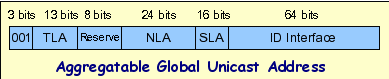
* Adresse de site local (Scope Site)

Elle peut être comparée aux adresses privées de l'IPv4. Elles peuvent être routées mais ne peuvent être utilisées sur internet. Le préfixe d'une adresse de site local est fec0::/10. Dans une adresse de ce type, après les 10 premiers bits, les 54 suivants définissent l'adresse de réseau. La RFC qui les a introduit a vu son statut passer en "deprecated" (2004 RFC 3879), ce qui sous-entend que chaque équipement aura son adresse publique. Nous pensions les adresses IPV6 locales passées aux oubliettes.

En 2005, la RFC 4193 introduit les adresses de type Unique Local Address. Le préfixe est fc00::/7 Le 8ième bit L est assignée à la valeur 1. Nous retrouvons ensuite un identifiant globale sur 40bits, un identifiant de sous réseau sur 16 bits et finalement l'interface sur 64bits. Cette RFC offre la possibilité de pouvoir utiliser un identifiant global permettant d'envisager l'interconnexion de sites sans devant changer l'identifiant des réseaux.

* Adresse utilisable sur Internet (Scope Global)

Ces adresses utilisent une représentation agrégée. Elles auront la structure suivante:



Le préfixe se retrouve codé sur trois bits. Actuellement la valeur est de 001 et correspond à une adresse unicast publique.

S'en suit le Top Level Agregator qui correspond aux différents blocs libérés par IANA et donc le premier porte le numéro 1. Les 16 bits correspondants seront alors 0010000000000001 et en hexadécimal, nous aurons 2001. Les 32 bits suivants dont les 8 de poids forts n'étaient pas encore affectés se retrouvent répartis de la façon suivante: 13 bits comme Sub-TLA et 19 bits comme NLA. Cette RFC est reléguée à l'heure actuelle comme archive et l'attribution des IP ne suit plus tout à fait cette règle.

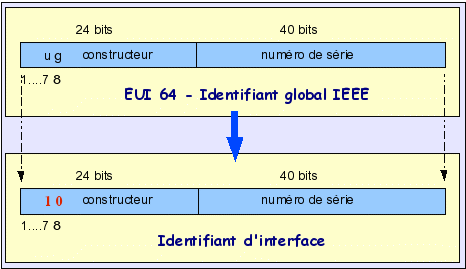
Les adresses IPV6 distribuées dans le premier bloc 2001:: aux RIR sont des blocs en /23. Les RIR distribueront les adresses IP aux LIR/Providers en blocs de /32

Belnet va donc recevoir une adresse en /32. Belnet disposera de 16 bits pour distribuer des adresses aux utilisateurs finaux et ce sont donc des adresses en /48 qui seront distribuées aux gros utilisateurs finaux. C'est la cas de l'implantation rue Frinoise qui a reçu de Belnet 2001:06a8:3480::/48. Sur une adresse composée de 128 bits, le site final aura à gérer 80 bits dont 16 seront utilisés pour identifiés un réseau et les 64 bits restant l'hôte sur ce réseau. Le site Rue Frinoise aura donc la possibilité d'interconnecter 2^16 réseaux et pour chaque réseau 2^64 hôtes. Ces adresses seront attribuées de façon automatique via un serveur DHCP ou de façon statique.

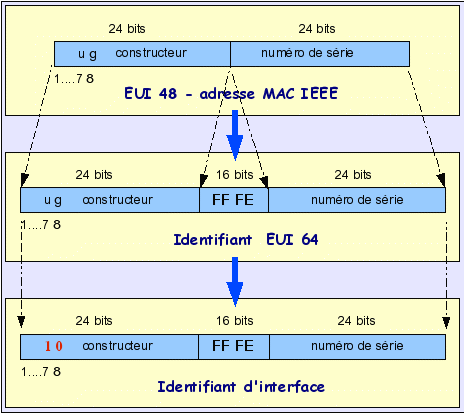
# Création des adresses de lien.

Nous avons précédemment parlé des adresses de lien local attribuées de façon automatique au démarrage de toute interface. Nous allons analyser la façon dont les 64 bits de l'ID interface seront choisis.

La première solution est de se baser sur une adresse physique qui serait dans un format EUI-64 de l'IEEE. Il suffira alors de simplement insérer cette adresse physique dans les 64 bits de l'adresse IPv6



La deuxième solution repose sur les adresses physiques représentées sur 48 bits comme le sont les adresses MAC de nos cartes réseau Ethernet. Il s'agit du format EUI-48 de l'IEEE. Pour combler les 64 bits de l'adresse IPv6, l'opération suivante sera alors effectuée



# Les adresses anonymes.

Microsoft a décidé que les adresses fabriquées à partir de l'EUI-64 n'était plus "anonyme" (on connaît l'adresse MAC donc le constructeur, et donc très virtuellement et potentiellement, éventuellement le revendeur). Donc sous XP/2003, par défaut (en présence d'un routeur annonçant un réseau), l'adresse normale est bien auto configurée mais une autre adresse l'est également, une adresse dite "anonyme" dont la partie hôte (en principe ce qui devait être l'EUI-64) est choisie au hasard. Il s'agit d'une fonction relativement grotesque, et qui est désactivable.

Pour chaque préfixe, chaque interface reçoit une ipv6 supplémentaire:   
**<prefixe><EUI-64>**

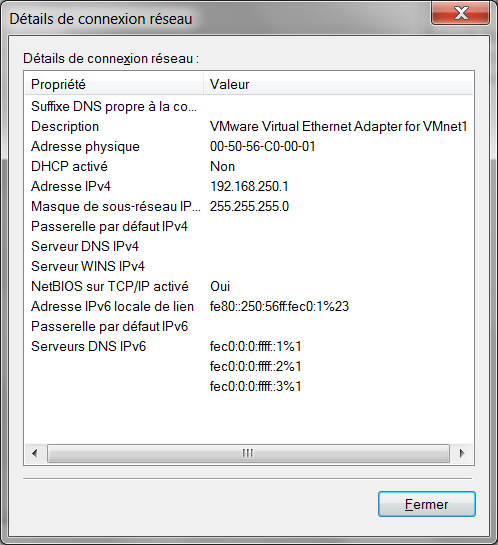
XP/2003, en plus, ajoutent également une ipv6 anonyme:  
**<prefixe><64 bits au hasard>**

Pour les connexions sortantes il utilise par défaut cette IP "anonyme".  
Pour les connexions entrantes, on peut le joindre sur les 2 IPs bien entendu.

Or, il a été vu que quel que soit le réseau de la machine, l'EUI-64 reste toujours le même. Donc quel que soit le réseau de la machine et la période de connexion, on peut l'identifier de façon unique.

L'adresse anonyme est censée palier ce que Microsoft a estimé être un problème, mais qui n'en est pas vraiment un, l'anonymat en matière de réseau étant finalement largement théorique.

Sous Windows seven, pour une pseudo interface utilisée par le logiciel Vmware.



Nous pouvons en vérifiant la valeur de l'adresse MAC de la carte réseau 00-50-56-C0-00-01, vérifier la création de l'adresse de lien

00-50-56 FFFE C0-00-01

Nous obtiendrons alors: **02**50:56FF:FEC0:0001 (bit 'u' positionné à 1)

L'adresse de lien locale sera donc bien fe80:250:56ff:fec0:1%23 (le %23 représente l'interface associée à cette adresse)

# Solicited Node Address, Neighbor Discovery, Router Advertissement

Pour chacune de ses adresses IPv6 (y compris la première d'entre elles, l'adresse de lien local de l'interface fabriquée avec l'EUI-64), chaque interface va adhérer à un groupe multicast bien particulier calculé en fonction de celle-ci.  
Cette adresse est la solicited node address. C'est une adresse multicast de lien local se présentant sous la forme:

**FF02:0:0:0:0:1:FFXX:XXXX**



Les X représentent la partie variable de l'adresse multicast. Cette partie contient les 24 derniers bits de l'adresse IPv6 d'origine (c'est à dire les 3 octets ou les 6 nibbles de droite dans l'adresse).  
Par conséquent, si seul l'adressage à base d'EUI-64 est utilisé, toutes les IPv6 de l'interface donnent lieu à l'adhésion à une seule solicited node address (car l'EUI-64 ne change pas même si la partie réseau de l'adresse change).

En connaissant l'IPv6, il est immédiatement possible de connaître la solicited node address relative à l'IPv6 considérée. Pour découvrir l'adresse MAC relative à une adresse IPv6 connue, il suffit d'envoyer un message ND (Neighbor Discovery) vers cette adresse en utilisant sa propre adresse de lien local.

**Intêret de l'utilisation de la solicited node address**: Par rapport au bon vieux broadcast, cette méthode fait appel à de multiples groupes multicast dont peu de nodes font partie en même temps. Il est bien entendu que le multicast sur un segment Ethernet de base se traduit par du broadcast au niveau 2. Cependant, dans le cadre d'équipements de niveau 2 et 3 gérant effectivement le multicast (chaque adhésion multicast IPv6 donne lieu à une adhésion au niveau 3, par ND, mais également au niveau 2 par du broadcast Ethernet), l'utilisation du ND et donc du multicast permettrait de réduire de façon spectaculaire le nombre de broadcasts sur le réseau, et de limiter le trafic de résolution des adresses MAC de façon très considérable, en tous cas de le convertir de broadcast à quasi-unicast.

Au démarrage, une interface a donc une adresse de lien local (avec EUI-64) et adhère à la solicited node address correspondante. Si aucun routeur n'est présent sur le lien, il n'y a pas d'autre adresse attribuée automatiquement. En effet le node va cherche la présence d'un routeur par le biais d'un multicast approprié: le Router Solicitation.

Si un routeur est présent et configuré, il va répondre à ce message (toujours multicast) par un RA (Router Advertisement). Puisque ce message a été expressément demandé, il s'agira d'un Solicited Router Advertisement. Il contient uniquement la ou les préfixes disponibles sur le réseau au format CIDR, et sa propre adresse. Connaissant l'adresse du réseau, la machine peut accoler le réseau et son EUI-64 pour former une adresse IPv6 routable et unique. Le routage est possible puisque la machine utilisera l'adresse du routeur reçue dans le RA. Si plusieurs routeurs sont présents, plusieurs adresses seront ainsi fabriquées et plusieurs routes par défaut rajoutées.

L'adresse est donc fabriquée de la sorte:  
**<64 bits de réseau, annoncés par le routeur sur le LAN><EUI-64: 64 bits hôtes, définis à partir de la MAC>**

Par ailleurs, tout routeur correctement configuré émet régulièrement sur le réseau des RA pour annoncer ces informations. Puisque personne n'a réclamé ces RA, il s'agit d'unsolicited Router Advertisements.

Bien entendu, il est possible de configurer des adresses statiques et indépendantes de l'EUI-64, ainsi que de désactiver l'auto configuration des adresses et de la route par défaut. C'est pratiquement indispensable sur les routeurs.

Lorsqu'un hôte a détecté qu'un routeur est présent, une route par défaut sera automatiquement ajoutée. Cette adresse est une adresse de lien et non pas une adresse globale. Ce choix est justifié par le fait qu'une adresse de lien ne change pas tandis qu'une adresse globale pourrait être changée par l'administrateur du réseau.

Voici un exemple de création de la Solicited Node Address.

Imaginons que nous ayons deux hôtes sur un segment Ethernet, l'hôte A et l'hôte B. L'hôte A a comme adresse de lien local fe80:100:100:100::A/64 tandis que l'hôte B a comme adresse de lien local fe80:100:100:100::B/64. L'hôte A désire communiquer avec l'hôte B.

1) L'hôte A doit connaître l'adresse physique de l'hôte B. Sous IPv4, cette récupération s'effectue au moyen du protocole ARP qui se base sur du broadcast. L'hôte A est capable d'envoyer un paquet vers l'adresse de multicast à laquelle l'hôte B est joint. Il suffira pour l'hôte A de prendre le préfixe **ff02:0:0:0:0:1:ff00::/104** et d'ajouter les 24 bits de poids faible de l'adresse de lien de l'hôte B, ce qui correspondra à **ff02:0:0:0:0:1:ff00:000B**

2) L'hôte B est déjà configuré au niveau de son adresse solicited-node et a rejoint le groupe **ff02:0:0:0:0:1:ff00:000B** en utilisant MLD ( remplacement IPv6 de IGMP). L'hôte B, comme membre de ce groupe, recevra un Neighbor Solicitation de l'hôte A.

––

3) L'hôte B répondra avec son adresse MAC dans la partie payload du Neighbor Advertisement message

Sous Windows Seven, pour la même pseudo interface Vmware, nous allons utiliser la commande permettant de visualiser les différents groupes multicast associés: netsh int ipv6 show joins. Nous obtenons alors l'affichage suivant:

Interface 23 : VMware Network Adapter VMnet1

Étendue Références Dern Adresse

---------- ---------- ---- -------------------

0 0 Oui ff01::1

0 0 Oui ff02::1

0 3 Oui ff02::c

0 1 Oui ff02::1:3

0 1 Oui **ff02::1:ffc0:1**

Nous pouvons constater, comme anoncé dans le paragraphe traitant des adresses multicastes, que notre interface se retrouve dans des groupes prédéfinis.

ff01::1 correspond à toutes les interfaces de notre hôte

ff02:1 correspond à tous les noeuds du lien

ff02:c: correspond à un protocole déprécié mais utilisé dans le protocole UPNP

ff02::1:3 correspond au protocole LLMNR (Link Local Multicast name Resolution). Permet la demande de résolution de nom au niveau des clients sans devoir disposer d'un serveur DNS. Attention: gourmand en ressource réseau, et il est conseillé de désactiver ce protocole.

ff02::1:ffc0:1: correspond à la sollicited node address.

# Attribution des adresses IP globales.

L'auto configuration des adresses en mode stateless a déjà été abordée dans le paragraphe précédent pour les adresses de lien locales. Ce mécanisme permettra aussi l'auto configuration des adresses globales au travers des échanges de messages de type Router Solicitation et Router Advertissement avec les routeurs voisin.

En stateless, nous pouvons configurer les adresses de façon automatique au travers de la découverte des préfixes (RFC 2462). Pour chacun des préfixes renseignés sur le routeur pour une interface réseau donnée, nous retrouverons deux flags que sont le flag 'Autonomous' et 'OnLink'.

**Autonomous:** Permet d'instruire l'hôte recevant cette donnée de générer une adresse de façon automatique à partir de ce préfixe. Si ce flag est à false, aucune adresse ne sera générée sur base de ce préfixe.

**Onlink:** Permet d'instruire l'hôte recevant cette donnée que le préfixe spécifié est 'onlink' et qu'il doit donc être utilisé dans la configuration automatique, notamment dans les tables de routage de l'hôte.

Dans cette solution, nous pouvons remarquer qu'aucune configuration n'est envisagée pour les mécanismes de résolution de nom internet. Pour ce faire, il faudra soit configurer ces informations manuellement, soit utiliser un serveur DHCPV6 qui se limitera dans ce rôle d'attribution des adresses IPV6 en stateless, d'attribuer des options telles que serveur DNS, serveur de temps...

Dans la configuration des messages de type Router advertissement sur un routeur, nous allons retrouver les deux flags Managed Configuration et Other Parameters Configuration qui permettront de définir la façon dont les adresses et paramètres puissent être obtenus de façon automatique d'un serveur DHCPV6. Ces deux flags sont également appelés M flag et O flag. Voici un résumé de l'effet de ces flags envoyés d'un routeur vers un hôte.

**Les flags M et O Flags sont à 0.** Cette combinaison correspond à un réseau sans serveur DHCPV6. Les paramètres devront être définies manuellement.

**Les flags M et O Flags sont à 1.** DHCPv6 est utilisé pour fournir les paramètres et l'adresse IPV6. Cette configuration est connue sous le nom DHCPV6 statefull.

**Le Flag M est à 0 and le Flag O est à 1.** DHCPv6 est utilisé pour assigné les paramètres tandis que l'adresse IPV6 est assignée automatiquement par les mécanismes de découverte des routeurs voisins. Cette combinaison est connue sous le nom stateless DHCPV6.

**Le Flag M est à 1 and le Flag O est à 0.** Dans cette combinaison, l'adresse est attribuée par le serveur DHCPV6 tandis que les paramètres ne sont pas définis.

Vous pouvez vous retrouver avec deux adresses IPV6 pour le même préfixe. Si vous définissez sur le routeur un préfixe avec le flag 'autonomous' tandis que le flag 'M' est mis à 1, deux adresses seront générées. L'une fournie par le serveur DHCPV6 et l'autre pour le préfixe fourni. Voici une copie d'écran d'une telle configuration sous Windows Seven...

# Les tables de routage sur les hôtes.

Dans les serveurs DHCPV4, il était possible de pouvoir définir des routes et une passerelle par défaut comme options à fournir sur les clients. Dans les premières normes de la version DHCPV6, ces options n'étaient pas présentes. A l'heure actuelle, ces options ont été ajoutées mais pas encore implémentées dans tous les serveurs et/ou clients DHCP.

Comme nous l'avons vu précédemment, la passerelle par défaut est ajoutée automatiquement lors de la découverte des routeurs. Pour ce qui est des tables de routage, elles seront générées sur base des préfixes définis sur les routeurs. Il suffira pour chacun des préfixes, de mettre le flag 'autonomous' à 0 et le flag 'OnLink' à 1.

Si plusieurs routeurs sont présents sur le même segment, nous pourrons définir un ordre de préférence pour les différentes routes fournies si des routes pouvaient être redondantes. L'hôte pourra donc voir un routeur comme routeur primaire ou routeur secondaire. (RFC 4191). Nous pouvons donner trois valeurs: High, Medium, Low

Pour la passerelle par défaut, il est possible si plusieurs routeurs existent de pouvoir supprimer la passerelle de l'hôte pour un routeur déterminé. Il suffira pour le routeur en question de placer le default lifetime à 0. Un hôte recevant cette propriété écartera l'adresse de lien du routeur si le lifetime vaut 0.

Pour nous aider dans les tests de configuration automatique dans le cas où un routeur serait présent sur le lien local, nous allons configurer une solution Linux Debian en y installant le package radvd. Voici la configuration du routeur (fichier radvd.conf dans /etc)

interface eth1 {

AdvSendAdvert on;

MinRtrAdvInterval 3;

MaxRtrAdvInterval 10;

AdvDefaultLifetime 100;

prefix 2001:06a8:3480:4::/64 {

AdvOnLink on;

AdvAutonomous on;

AdvRouterAddr off;

};

};

Une fois le service démarré, nous obtenons les informations suivantes dans la pseudo interface côté Windows.

Carte Ethernet VMware Network Adapter VMnet1 :

Suffixe DNS propre à la connexion. . . :

Adresse IPv6. . . . . . . . . . . . . .: 2001:6a8:3480:4:250:56ff:fec0

Adresse IPv6 de liaison locale. . . . .: fe80::250:56ff:fec0:1%23

Adresse IPv4. . . . . . . . . . . . . .: 192.168.250.1

Masque de sous-réseau. . . . . . . . . : 255.255.255.0

Passerelle par défaut. . . . . . . . . : fe80::20c:29ff:feb1:15cc%23

L'adresse de liaison locale est toujours bien présente. Nous retrouvons en plus une adresse générée automatiquement sur base du préfixe renseigné sur le routeur. Cette génération est rendue possible du fait du flag 'AdvAutonomous' positionné à 1. Au préfixe représenté sur 64bits sera ajouté l'EUI-64 (identique à celui utilisé pour la création de l'adresse de lien)

Une adresse de passerelle est ajoutée. Elle correspond à l'adresse de liaison locale du routeur.

eth1 Link encap:Ethernet HWaddr 00:0c:29:b1:15:cc

adr inet6: fe80::20c:29ff:feb1:15cc/64 Scope:Lien

Si nous analysons la table de routage, nous obtenons les données suivantes.

If Metric Network Destination Gateway

23 276 ::/0 **fe80::20c:29ff:feb1:15cc**

1 306 ::1/128 On-link

23 28 **2001:6a8:3480:4::/64** On-link

Nous retrouvons la route par défaut évoquée juste avant et correspondant à l'adresse de liaison du routeur. Nous retrouvons également la route associée au préfixe fourni par le routeur. Si plusieurs routeurs sont présents et que l'on souhaite supprimer une route par défaut pour l'un d'entre eux, il est possible de supprimer cette route en définissant le temps de vie de l'information fournie par le routeur à 0.

interface eth1 {

AdvSendAdvert on;

MinRtrAdvInterval 3;

MaxRtrAdvInterval 10;

**AdvDefaultLifetime 0;**

prefix 2001:06a8:3480:4::/64 {

AdvOnLink on;

AdvAutonomous on;

AdvRouterAddr off;

};

};

Carte Ethernet VMware Network Adapter VMnet1 :

Suffixe DNS propre à la connexion. . . :

Adresse IPv6. . . . . . . . . . . . . .: **2001:6a8:3480:4:250:56ff:fec0:1**

Adresse IPv6 de liaison locale. . . . .: fe80::250:56ff:fec0:1%23

Adresse IPv4. . . . . . . . . . . . . .: 192.168.250.1

Masque de sous-réseau. . . . . . . . . : 255.255.255.0

**Passerelle par défaut. . . . . . . . . :**

L'adresse dans le scope global est toujours présente mais la passerelle par défaut est maintenant absente.

Si nous souhaitons pouvoir ajouter des routes supplémentaires associées à d'autres préfixes sans nécessairement avoir une génération automatique d'adresse basée sur ce dernier, nous pouvons jouer sur le flag AdvAutonomous que nous placerons à 0. Exemple:

interface eth1 {

AdvSendAdvert on;

MinRtrAdvInterval 3;

MaxRtrAdvInterval 10;

AdvDefaultLifetime 0;

prefix 2001:06a8:3480:4::/64 {

AdvOnLink on;

AdvAutonomous on;

AdvRouterAddr off

};

**prefix 2001:06a8:3480:5::/64 {**

**AdvOnLink on;**

**AdvAutonomous off;**

**AdvRouterAddr off;**

**};**

};

Si nous regardons les adresses automatiques générées, nous remarquerons qu'aucune adresse automatique n'a été générée pour le préfixe 2001:06a8:3480:5::/64 mais nous retrouverons cette adresse dans les tables de routage.

23 28 2001:6a8:3480:4::/64 On-link

23 28 2001:6a8:3480:5::/64 On-link

Descriptif des paramètres