

# Prototype dual-stack IPv4/6 sur un backbone MPLS-VPN (services et sécurité)

Projet de Bachelor – Télécommunication

**Steve Lienhard**

Professeur responsable : M. Stephan Robert, HEIG-VD

Mandant : M. Jérôme Vernez, SIEN



**Juillet 2011**

## **TABLE DES MATIERES**

<b>1</b>	<b>RÉSUMÉ .....</b>	<b>3</b>
<b>2</b>	<b>INTRODUCTION .....</b>	<b>5</b>
<b>3</b>	<b>ADRESSAGE IPV6 .....</b>	<b>7</b>
3.1	ADRESSES « LINK LOCALES » .....	7
3.2	ADRESSES DE SITE .....	9
3.3	ADRESSES GLOBALES.....	10
3.4	ADRESSES MULTICAST .....	10
3.5	ADRESSES ANYCAST .....	10
<b>4</b>	<b>PROTOTYPE .....</b>	<b>11</b>
<b>5</b>	<b>SERVICES DE BASE.....</b>	<b>13</b>
5.1	ICMPv6.....	13
5.2	DHCPv6 .....	21
5.3	DNS .....	36
<b>6</b>	<b>SECURITE .....</b>	<b>43</b>
6.1	TOOLKIT.....	44
6.2	WINDOWS DENIAL OF SERVICE.....	44
6.3	MAN IN THE MIDDLE .....	46
6.4	SCANNING .....	48
6.5	FLOODING .....	49
6.6	DOS-NEW-IP .....	51
6.7	DHCP SNOOPING .....	53
6.8	SECURE NEIGHBOR DISCOVERY .....	55
6.9	CISCO ASA.....	57
6.10	MICROSOFT DIRECTACCESS .....	59
<b>7</b>	<b>CONCLUSION .....</b>	<b>62</b>
<b>8</b>	<b>GLOSSAIRE.....</b>	<b>65</b>
<b>I.</b>	<b>SOURCES .....</b>	<b>67</b>
I.	BIBLIOGRAPHIE .....	67
II.	WEBGRAPHIE .....	67

<b>A.</b>	<b>ANNEXES .....</b>	<b>70</b>
A.	CONFIGURATION DHCPv6 .....	70
B.	CONFIGURATION DNS .....	73
C.	MISE EN PLACE D'UNE INFRASTRUCTURE ASA/AD.....	82
D.	INSTALLATION DIRECT ACCESS .....	85
E.	WEB ENROLLEMENT .....	91
F.	JOURNAL DE TRAVAIL .....	94

## 1 Résumé

Ce travail réalisé dans le cadre du diplôme de Bachelor de l'orientation « Télécommunication Réseaux et Services » de la Haute d'Ecole d'Ingénierie et Gestion du Canton de Vaud (HEIG-VD), et mandaté par le service informatique de l'entité neuchâteloise (SIEN), a pour but d'étudier la nouvelle norme d'adresses IP appelée IPv6, amenée à remplacer l'actuelle IPv4. Cette étude a comme principal objectif de donner un aperçu des nouveaux concepts apportés par la norme par rapport à l'ancienne, tout en restant basée sur le réseau actuel du SIEN et de ses équipements. Pour se faire, un prototype a été conçu afin de représenter au mieux l'infrastructure mise en place à l'heure actuelle par le SIEN. Celui-ci permet un rapprochement entre les normes/standards présentés et la réalité du terrain mise en valeur tout au long du rapport. La présentation est essentiellement basée sur la pratique des différentes notions proposées par IPv6, plutôt que d'entrer dans des détails théoriques n'étant pas primordiaux dans la mise en place des services fournis par le SIEN.

Après un bref rappel des concepts d'adressage, qui diffèrent passablement de l'ancienne norme, ce rapport présente le protocole ICMPv6 et ses nombreuses fonctionnalités, se rapportant au « Neighbor Discovery Protocol ». Toutes ces fonctions sont présentées par rapport à un schéma de référence, lui-même faisant partie du prototype de réseau principal afin d'offrir au lecteur une meilleure compréhension du protocole. Lorsque nécessaire, une distinction est faite entre les cas où les machines sont dans le même sous-réseau et ceux où elles ne le sont pas, afin de traiter en détail les différentes possibilités. La partie traitant d'ICMPv6 contient également un chapitre présentant la découverte de la MTU sur un lien local et global.

La suite de ce chapitre traite des deux autres services de base les plus utilisés dans les réseaux actuels, à savoir le DHCP et le DNS. Il est donc présenté tout d'abord les nouvelles fonctionnalités qu'offre le service DHCPv6 par rapport à son prédécesseur, leurs implications, leurs avantages et surtout dans quels cas ils sont utilisables. La partie DNS comporte une étude en profondeur du service dans le cadre d'IPv6, avec des mises en pratiques basées sur des résolutions de noms effectuées sur un serveur Windows 2008.

Le chapitre suivant traite de la sécurité et présente en premier lieu les différentes failles de sécurité présentes dans la norme, afin de sensibiliser le lecteur aux vulnérabilités potentielles qu'il est possible d'exploiter dans le cadre d'activités illégales. Ces différentes failles sont toutes réalisées et présentées dans ce rapport, afin de démontrer la simplicité de mise en œuvre de telles attaques, et donc leur dangerosité. Les contremesures à appliquer sont bien évidemment décrites ensuite afin de pouvoir mettre en place les différents mécanismes garantissant un minimum de sécurité dans des réseaux basés sur IPv6.

Cette partie sécurité présente aussi deux produits utilisés par le SIEN, à savoir la « gateway/firewall » ASA de Cisco ainsi que le « Direct Access » de Microsoft, afin d'une part de tester la compatibilité avec IPv6 des fonctionnalités qu'ils offrent et d'autre part de montrer les avantages à les utiliser dans le cadre de la nouvelle norme.

Enfin, toutes les manipulations présentées dans ce rapport sont reproductibles grâce à la partie « Annexe » qui contient toutes les configurations effectuées.

Il est important de mentionner que ce travail est basé sur une philosophie de migration « dual-stack » en vue d'une migration douce où il est question de faire cohabiter IPv4 et IPv6 ensemble afin de petit à petit passer de l'ancienne à la nouvelle norme. La problématique d'une migration utilisant d'autres concepts comme le tunneling n'est donc pas abordée.

Ce travail doit également être considéré comme un complément à celui réalisé par M. Julien Tissot, dans le sens où celui-ci traite les impacts du côté « services et applications » de la norme, par rapport au sien qui traite le côté « réseau ».

## 2 Introduction

Dans le monde d'aujourd'hui, rares sont les appareils dans notre vie de tous les jours qui ne sont pas reliés à internet. Du plus banal des ordinateurs, au plus folklorique des frigos, en passant bien évidemment par le Smartphone ou la télévision, de plus en plus de ces éléments sont capables de communiquer à travers ce gigantesque réseau. Pour donner un ordre de grandeur, les 61 millions d'utilisateurs connectés en 1996 se sont décuplés en 6 ans pour atteindre quelques 604 millions en 2002 et ces chiffres restent pour le moins raisonnables comparés aux 1.9 milliards de personnes que représente actuellement la communauté internet. Ces statistiques exorbitantes font immédiatement apparaître une problématique évidente : celle d'un espace d'adressage trop petit pour supporter un tel essor.

En effet, les concepteurs initiaux du réseau ARPA (l'ancêtre de l'actuel « world wide web ») étaient loin d'imaginer que leur réseau, initialement destiné à connecter plusieurs universités entre elles, allait devoir supporter plus d'un milliard d'utilisateurs. Ainsi, les adresse IPv4 en fonction actuellement codées sur 32 bits (ce qui représente un total arrondi de 4.3 milliard d'adresses disponibles) ne suffisent plus à un tel point qu'il n'existe à l'heure actuelle que très peu d'adresses encore disponibles. Malgré des technologies comme le NAT, qui n'a fait que repousser le problème, il est maintenant nécessaire de se concentrer sur l'avenir en anticipant le nombre croissant d'internautes. C'est à partir de là qu'intervient la nouvelle norme d'adresse nommée « IPv6 », dont les buts sont multiples. En plus d'agrandir le nombre total d'adresses possibles, grâce à un codage sur 128 bits, elle a été conçue pour résoudre certains problèmes de sécurité présents dans l'ancienne norme, mais également pour tenter un routage plus efficace des paquets.

Cette « technologie » récente comporte bien sûr différents avantages et inconvénients, dont il est indispensable de s'informer. De nos jours, la nécessité pour les administrateurs réseau de s'y intéresser est bien présente et ceci pour plusieurs raisons. Outre le fait que de moins en moins d'adresses sont disponibles en IPv4, certains services, notamment le « Direct Access » de Microsoft, sont compatibles uniquement en IPv6. En plus de cela, des pays pionniers en la matière, comme certains pays asiatiques comme l'Inde ou la Chine sont pour la plupart nettement en avance sur les autres par rapport à la nouvelle norme. Ceci implique que de plus en plus de services qu'ils proposent ne vont plus être compatibles avec l'ancienne. En plus de cela, un bon nombre des clients potentiels ne seront, dans un avenir proche, plus atteignables via IPv4.

La question du coût de migration très élevé est trop souvent un prétexte pour ne pas prêter attention à la migration vers IPv6. Lors de ce calcul, il est nécessaire de tenir compte de l'investissement dans des infrastructures dont la durée de vie reste limitée (la norme IPv4 étant amenée à disparaître), par rapport à un investissement sur le long terme, sur une norme émergente qui va probablement être utilisée pendant de très

nombreuses années.

Grâce à tous ces arguments, on constate clairement qu'il est maintenant temps pour toute entreprise de se projeter vers le futur et de s'intéresser à IPv6. Il est évident qu'une migration de ce type comporte énormément de facteurs à prendre en compte, de tests à effectuer, et donc de temps investis. C'est pour cela que différents concepts de migration existent, comme le dual-stack (sur lequel est basé ce travail) ou le tunneling. Ces deux architectures permettent de passer d'une norme à l'autre « en douceur » afin de tester la compatibilité de chaque produit. Dans le cadre de cette migration, il est aussi important de sensibiliser les ingénieurs à la façon dont réaliser de nouveaux produits qui sont directement compatibles avec la nouvelle norme, afin d'éviter des coûts supplémentaires par la suite.

Malgré un effort important à considérer, il est donc temps de réfléchir en terme d'IPv6 tout autant que d'IPv4 pour chaque nouveau produit, chaque nouveau service, et chaque nouveau réseau mis en place actuellement, afin d'être un précurseur en la matière et de pouvoir effectuer une migration lente et réfléchie permettant de minimiser les risques au maximum.

### 3 Adressage IPV6

Comme bref rappel, il peut être tout d'abord utile de rappeler les différentes notions d'adressage introduites par la norme IPv6, puisque celles-ci diffèrent considérablement par rapport à son homologue IPv4. Pour rappel, en IPv4 nous pouvons classer les adresses selon les catégories suivantes (chacune des adresses pouvant faire partie de plusieurs de ces catégories) :

- privée (non-routable sur Internet)
- globale (routable sur Internet)
- unicast
- multicast
- broadcast

En IPv6, d'autres catégories ont été définies, qui sont chacune détaillée dans la partie qui suit :

- link locale
- globale
- unicast
- multicast
- anycast

#### 3.1 Adresses « link locales »

Ce type d'adresse, comme son nom l'indique, peut être utilisé sur un lien local uniquement, c'est-à-dire qu'elles ne traversent aucun routeur. Ces adresses restent donc à l'intérieur d'un sous-réseau spécifié. Les adresses de lien local sont utiles principalement au protocole « Neighbor Discovery » qui permet la configuration de l'adresse globale d'une interface. Ce protocole sera expliqué en détail par la suite. Une connexion point à point entre deux routeurs peut également utiliser ce type d'adresse, sans que l'administrateur ne configure quoi que ce soit. Elle offre donc la possibilité à des machines connectées entre elles de pouvoir communiquer sur un lien local, sans pour autant qu'une configuration préalable ait eu lieu. Une adresse de ce type est simplement construite en concaténant le préfixe réseau « **FE 80:: /64** » aux 64 bits (EUI-64) de l'identifiant de l'interface. Elles peuvent bien sûr être configurées manuellement si l'administrateur le désire.

Une adresse EUI-64 est obtenue selon le schéma de la Figure 1.



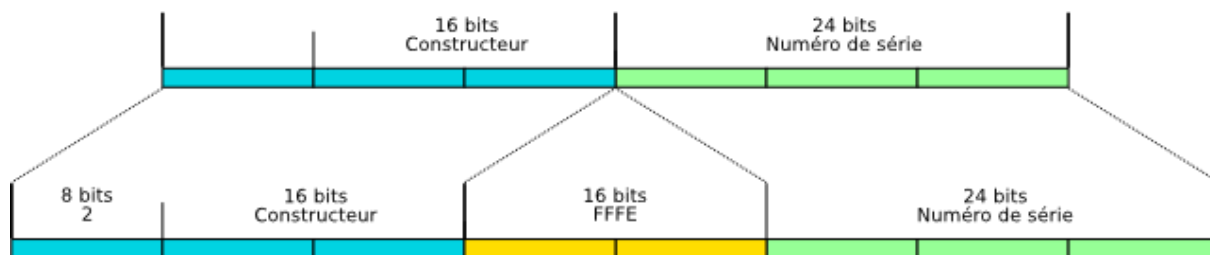


Figure 1 : Construction d'une adresse link locale selon la norme EUI-64.

(Source : <http://www.isinax.info>)

Pour obtenir une adresse de ce type, il suffit de prendre l'adresse MAC de l'interface (composée de 48 bits) d'y ajouter 16 bits (fffe) entre les 16 bits du constructeur et les 24 du numéro de série. Les 8 bits de début sont utilisés selon la Figure 2.

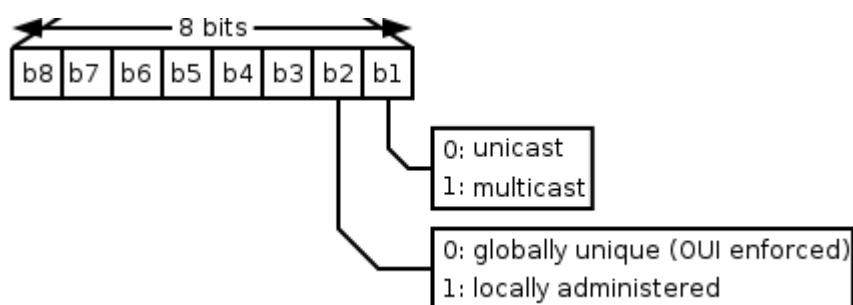


Figure 2 : Signification des 8 premiers bits d'une adresse EUI-64 bits.

(Source : Wikipédia)

Le premier bit est utilisé pour indiquer si l'adresse locale est une adresse unicast ou multicast, et le second pour indiquer si elle est globalement unique (donc si l'adresse IP est construite grâce à l'adresse MAC) ou administrée localement. A noter que les adresses locales sous Windows 7 ne respectent pas cette norme, mais sont choisies par l'hôte d'une manière pseudo-aléatoire, et n'ont donc aucun lien avec l'adresse MAC de la carte réseau. Les adresses locales sous Windows XP suivent par contre la logique « EUI-64 ». Ces machines peuvent utiliser ensuite le protocole NDP (présenté dans la partie « ICMPv6 ») afin de s'assurer qu'aucune duplication d'adresse locale n'est présente sur le LAN.

La Figure 3 est un exemple d'une adresse locale de lien prise sous Windows 7.

```

Carte Ethernet Connexion au réseau local :
  Suffixe DNS propre à la connexion. . . : local.neuch
  Description. . . . . : Intel(R) 82577LM Gigabit Network Con
  nection
  Adresse physique . . . . . : 88-AE-1D-AF-4E-50
  DHCP activé. . . . . : Oui
  Configuration automatique activée. . . : Oui
  Adresse IPv6. . . . . : 2001:4da0:c00:0:bf27:8de5:aded:f629<
  préfére)
  Bail obtenu. . . . . : mercredi 25 mai 2011 10:10:15
  Bail expirant . . . . . : vendredi 27 mai 2011 10:10:15
  Adresse IPv6 de liaison locale. . . . . : fe80::e03d:e482:5f46:92c7%13 préfére)
  Passerelle par défaut. . . . . : fe80::226:52ff:fe96:2443%13
  IAID DHCPv6 . . . . . : 294170141
  DUID de client DHCPv6. . . . . : 00-01-00-01-14-FC-3B-2D-88-AE-1D-AF-4E
  -50
  Serveurs DNS. . . . . : 2001:4da0:c01:0:d939:58f5:1798:5787
  NetBIOS sur TCP/IP. . . . . : Désactivé
  Liste de recherche de suffixes DNS propres à la connexion :
    local.neuch
  
```

Figure 3 : Adresse locale « pseudo aléatoire » construite sous Windows 7.

On constate clairement le préfixe « fe80 » (propre à ce type d'adresse) ainsi que la différence entre l'adresse MAC et l'adresse IPv6. Si maintenant la même capture est prise sous Linux, il apparaît cette fois-ci que l'adresse de liaison locale se crée à partir de l'adresse physique (Figure 4).

```

[steve@steve-laptop ~]$ ifconfig
eth0      Link encap:Ethernet  HWaddr 88:AE:1D:AF:4E:50
          inet6 addr: fe80::8aae:1dff:feaf:4e50/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:4 errors:0 dropped:0 overruns:0 frame:0
          TX packets:11 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:344 (344.0 b)  TX bytes:1038 (1.0 KiB)
          Interrupt:20 Memory:d7400000-d7420000
  
```

Figure 4 : Adresse locale construite selon la norme EUI-64 sous Linux.

L'adresse de lien est donc bel et bien construite avec l'adresse EUI-64 de la carte (« fffe » au milieu de l'adresse) concaténée au préfixe réseau « fe80 ».

## 3.2 Adresses de site

Les adresses de site permettent à des équipements de pouvoir communiquer **uniquement sur un site**. Elles ne sont donc pas routables sur Internet mais peuvent par contre, à l'inverse des adresses « link locales », traverser les routeurs. Cela permet par exemple à toutes les machines d'un site de communiquer sans pour autant devoir demander de préfixe au routeur. Ces adresses sont reconnaissables par le préfixe suivant : « fec0 :: /48 ».

Bien qu'ayant une utilité non négligeable, ce type d'adresse n'a pas vraiment réussi à s'imposer lors de la standardisation d'IPv6, à tel point qu'il risque d'être abandonné lors de l'écriture de la version finale de la RFC IPv6. Le choix reste tout de même possible pour des administrateurs réseau de les utiliser si cela rend plus simple leur architecture, notamment au niveau de l'adressage. On peut en effet imaginer des serveurs internes qui doivent être accessibles uniquement depuis l'intranet. Une adresse de site peut donc être très utile dans ce cas, puisqu'elle permettrait à la machine d'être atteinte par tout le réseau interne.

### 3.3 Adresses globales

Les adresses globales IPv6 sont similaires aux adresses globales IPv4. Celles-ci sont donc routables sur internet. Néanmoins, une des différences notables entre les deux normes est **qu'une seule interface peut avoir plusieurs adresses IPv6 et donc plusieurs adresses globales**. Une interface n'est donc pas liée à une adresse IP comme c'était le cas en IPv4. Par cette possibilité, une interface peut communiquer avec plusieurs réseaux à la fois, ce qui ouvre de nouvelles perspectives. Ces adresses ont le préfixe réseau suivant : « **2001::/16** ».

***Remarque :** Il existe d'autres préfixes possibles pour ce type d'adresse qui ne seront pas abordés dans ce travail, puisque ce dernier est basé sur un environnement de production. Les préfixes utilisés pour les réseaux expérimentaux (3FFE ::/16) ainsi que ceux utilisés pour la transition 6to4 (2002 ::/16).*

### 3.4 Adresses multicast

Les adresses multicast IPv6 ont la même utilité qu'en IPv4, c'est-à-dire pouvoir contacter un groupe défini de machines. Elles sont dérivées du préfixe réseau « **FF00 :: /8** ». Les adresses multicast sont typiquement utilisées par un client voulant contacter un serveur DHCP.

### 3.5 Adresses anycast

Enfin, les adresses anycast sont un tout nouveau type d'adresses introduit par la norme IPv6. Ces adresses permettent de router des données vers le service informatique « le plus proche ». Par exemple, si plusieurs serveurs DNS sont présents dans un réseau, la machine effectuant une requête de résolution de nom va pouvoir contacter le serveur DNS le plus proche d'elle. Mentionné autrement, ces adresses permettent d'envoyer un paquet à un ensemble de machines de destinations prédéfinies dans un groupe (correspondant à un service demandé comme le DNS par exemple) au lieu de l'envoyer à une interface déterminée afin de connaître

l'emplacement du service.

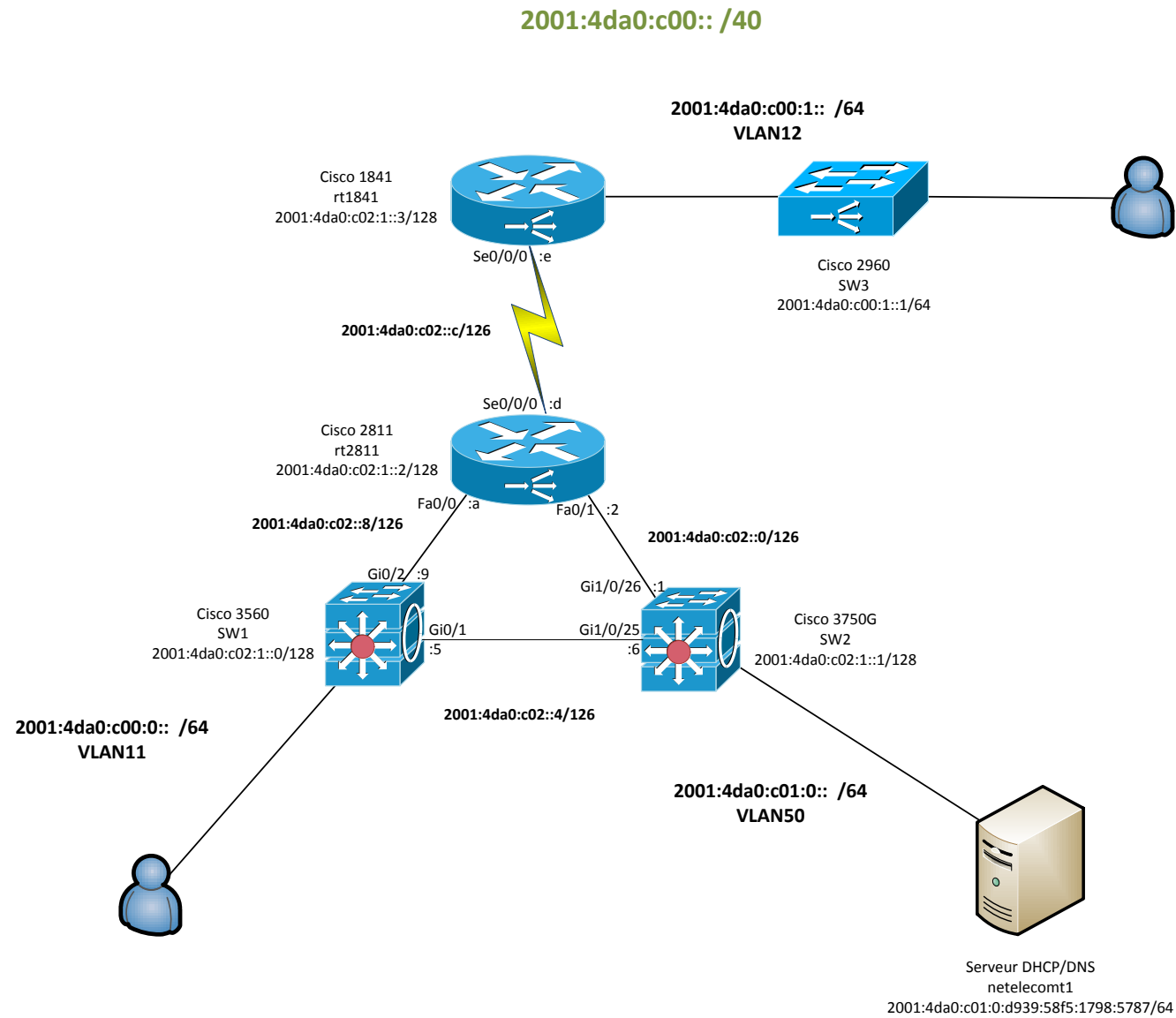
La différence entre les adresses anycast et les adresses multicast réside dans le fait qu'en multicast, un hôte envoie son paquet à l'adresse multicast du groupe et les machines abonnées à ce groupe vont toutes recevoir chaque paquet. En anycast par contre, la source va par exemple envoyer un message « Neighbor solicitation » pour déterminer l'adresse MAC de l'équipement, quatre serveurs vont répondre mais la source va en choisir un seul (le plus proche) pour effectuer ses requêtes.

## 4 Prototype

Avant d'aller plus loin, il est nécessaire de présenter le prototype faisant office de référence pour les tests qui vont suivre. Le schéma qui suit a été réfléchi afin de pouvoir mettre en pratique les notions découvertes lors de la phase de recherche. Le réseau se compose des éléments suivants :

- 1 serveur (Windows Server 2008) sur lequel est installé les services DNS et DHCPv6
- Plusieurs clients
- 3 routeurs
- 1 modem

Toutes les explications qui suivent sont basées sur le schéma suivant. Cependant, un autre schéma plus spécifique est souvent présenté afin de pouvoir cibler les éléments à prendre en compte.



## 5 Services de base

### 5.1 ICMPv6

Le protocole « ICMPv6 » est le point central de la nouvelle norme, puisque c'est lui qui régit toutes les interactions entre les machines du réseau local lorsque celles-ci se connectent. Ce protocole inclut le protocole NDP (Neighbor Discovery Protocol) qui propose les fonctionnalités suivantes :

- Traduction d'adresse physique en adresse IP (équivalent à l'ARP d'IPv4)
- Détection si un voisin est toujours actif sur le réseau
- Détection d'adresse dupliquée

D'autres fonctionnalités font également partie d'NDP mais ne sont pas présentées, celles-ci étant nettement moins courantes que les trois ci-dessus.

Une autre fonction d'ICMPv6 est la découverte de MTU, également présentée dans cette partie.

#### 5.1.1 Traduction d'adresses

La première fonction est le mécanisme d'ARP connu en IPv4. Grâce au NDP, ce dernier n'existe plus. A la place, c'est le protocole ICMPv6 qui est utilisé pour traduire les adresses IP en adresses MAC et inversement.

Pour illustrer cette fonctionnalité, différentes situations basées sur le prototype de réseau de la Figure 5 vont suivre.

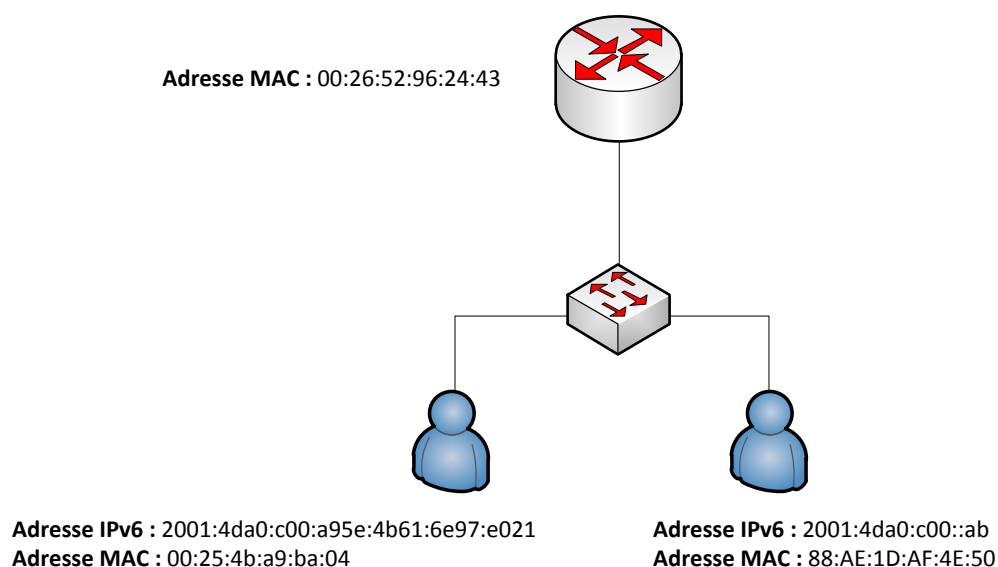


Figure 5 : Schéma du réseau local permettant d'illustrer la fonctionnalité « ARP » du protocole « Neighbor Discovery ».

### 5.1.1.1 Ping stateless local

Comme il sera expliqué plus en détail par la suite, le mode stateless impose au client de s'auto-configurer son adresse IPv6 et qu'il aille chercher les différents paramètres nécessaires sur le serveur. Dans cette configuration, le client reçoit du routeur son adresse de réseau ainsi que son préfixe. **Il sait donc dans quel sous-réseau il se trouve**, et si la destination s'y trouve également. La Figure 6 présente ce qui est obtenu lors d'un ping d'un client à un autre :

2001:4da0:c00:0:a95e:4b61:6e97:e021	ff02::1:ff00:ab	ICMPv6 Neighbor solicitation for 2001:4da0:c00::ab from 00:25:4b:a9:ba:04
2001:4da0:c00::ab	2001:4da0:c00:0:a95e:4b61:6e97:e021	ICMPv6 Neighbor advertisement 2001:4da0:c00::ab (sol, ovr) is at 88:ae:1d:af:4e:50
2001:4da0:c00:0:a95e:4b61:6e97:e021	2001:4da0:c00::ab	ICMPv6 Echo (ping) request id=0x0001, seq=68
2001:4da0:c00::ab	2001:4da0:c00:0:a95e:4b61:6e97:e021	ICMPv6 Echo (ping) reply id=0x0001, seq=68
2001:4da0:c00:0:a95e:4b61:6e97:e021	2001:4da0:c00::ab	ICMPv6 Echo (ping) request id=0x0001, seq=69
2001:4da0:c00::ab	2001:4da0:c00:0:a95e:4b61:6e97:e021	ICMPv6 Echo (ping) reply id=0x0001, seq=69
2001:4da0:c00:0:a95e:4b61:6e97:e021	2001:4da0:c00::ab	ICMPv6 Echo (ping) request id=0x0001, seq=70
2001:4da0:c00::ab	2001:4da0:c00:0:a95e:4b61:6e97:e021	ICMPv6 Echo (ping) reply id=0x0001, seq=70
2001:4da0:c00:0:a95e:4b61:6e97:e021	2001:4da0:c00::ab	ICMPv6 Echo (ping) request id=0x0001, seq=71
2001:4da0:c00::ab	2001:4da0:c00:0:a95e:4b61:6e97:e021	ICMPv6 Echo (ping) reply id=0x0001, seq=71

Figure 6 : Paquets échangés lors d'un ping effectué sur le réseau local de la Figure 5 lorsque le DHCP est en mode stateless ».

La machine source effectue un « Neighbor solicitation » qui permet de connaître l'adresse MAC de la machine de destination. C'est précisément ce paquet qui remplace l'ARP d'IPv4 comme on le constate sur la Figure 7.

```

Internet Control Message Protocol v6
  Type: 135 (Neighbor solicitation)
  Code: 0
  Checksum: 0x3bba [correct]
  Reserved: 0 (Should always be zero)
  Target: 2001:4da0:c00::ab (2001:4da0:c00::ab)
  ICMPv6 option (Source link-layer address)
    Type: Source link-layer address (1)
    Length: 8
    Link-layer address: 00:25:4b:a9:ba:04
  
```

Figure 7 : Analyse du paquet « Neighbor solicitation » tiré de la Figure 6.

Ensuite, le ping peut être envoyé comme il se doit à l'adresse IP voulue. Cette adresse est traduite en adresse MAC grâce au « Neighbor solicitation » précédent (Figure 6). En regardant la trame Ethernet à la Figure 8, on remarque que c'est bel et bien l'adresse physique de la machine de destination qui est mentionnée.

```

Ethernet II, Src: Apple_a9:ba:04 (00:25:4b:a9:ba:04), Dst: CompalIn_af:4e:50 (88:ae:1d:af:4e:50)
  Destination: CompalIn_af:4e:50 (88:ae:1d:af:4e:50)
  Source: Apple_a9:ba:04 (00:25:4b:a9:ba:04)
  Type: IPv6 (0x86dd)
  
```

Figure 8 : Analyse d'une trame tirée de la Figure 6 où l'adresse MAC de destination est celle de l'ordinateur de destination.

### 5.1.1.2 Ping statefull local

Cette fois-ci, la grosse différence avec le premier cas est que le client ne va plus configurer son adresse lui-même grâce au préfixe envoyé par le routeur mais va récupérer son adresse auprès d'un serveur DHCP. Ceci est également expliqué dans la partie « DHCPv6 ».

2001:4da0:c00::ab	2001:4da0:c00:0:d947:f330:c09c:30d1	ICMPv6 Echo (ping) request id=0x0001, seq=36
fe80::226:52ff:fe96:2443	ff02::1:ff9c:30d1	ICMPv6 Neighbor solicitation for 2001:4da0:c00:0:d947:f330:c09c:30d1
2001:4da0:c00:0:d947:f330:c09c:30d1	2001:4da0:c00::ab	ICMPv6 Echo (ping) reply id=0x0001, seq=36
2001:4da0:c00::ab	2001:4da0:c00:0:d947:f330:c09c:30d1	ICMPv6 Echo (ping) request id=0x0001, seq=37
2001:4da0:c00:0:d947:f330:c09c:30d1	2001:4da0:c00::ab	ICMPv6 Echo (ping) reply id=0x0001, seq=37
2001:4da0:c00::ab	2001:4da0:c00:0:d947:f330:c09c:30d1	ICMPv6 Echo (ping) request id=0x0001, seq=38
2001:4da0:c00:0:d947:f330:c09c:30d1	2001:4da0:c00::ab	ICMPv6 Echo (ping) reply id=0x0001, seq=38
2001:4da0:c00::ab	2001:4da0:c00:0:d947:f330:c09c:30d1	ICMPv6 Echo (ping) request id=0x0001, seq=39
2001:4da0:c00:0:d947:f330:c09c:30d1	2001:4da0:c00::ab	ICMPv6 Echo (ping) reply id=0x0001, seq=39

Figure 9 : Paquets échangés lors d'un ping effectué sur le réseau local de la Figure 5 lorsque le DHCP est en mode « statefull ».

**Remarque :** Par rapport au schéma de la Figure 5, l'adresse IPv6 d'une des machines a changé en 2001:4da0:c00:0:d947:f330:c09c:30d1

Ce mode de configuration implique également que **le client ne connaît pas son préfixe réseau** (puisque'il n'est plus transmis par le routeur). Il n'a donc aucun moyen de savoir s'il est dans le même sous-réseau que sa destination. Ainsi, il est obligé de passer par le routeur pour transmettre des trames locales comme le montre l'adresse MAC de destination de la Figure 10.

```

Ethernet II, Src: CompalIn_af:4e:50 (88:ae:1d:af:4e:50), Dst: Cisco_96:24:43 (00:26:52:96:24:43)
  Destination: Cisco_96:24:43 (00:26:52:96:24:43)
  Source: CompalIn_af:4e:50 (88:ae:1d:af:4e:50)
  Type: IPv6 (0x86dd)

```

Figure 10 : Analyse d'une trame tirée de la Figure 9 où l'adresse MAC de destination et celle du routeur.

L'adresse physique de destination montre qu'il ne s'agit plus cette fois de l'adresse de la machine à qui est destiné le ping mais bel et bien de l'adresse MAC du routeur.

Les deux cas de figure présentés dans cette partie montrent donc que la façon dont une machine effectue un ping sur le réseau local diffère en fonction du mode de DHCP utilisé, puisque dans un cas la machine connaît le préfixe de son réseau et est en mesure de savoir si la machine destinataire se situe dans le même sous-réseau qu'elle ou non, et dans l'autre elle ne le connaît pas et est donc obligée de passer par le routeur pour envoyer sa requête.



### 5.1.2 NUD

Une nouvelle fonction apportée par ICMPv6 est le NUD (Neighbor Unreachability Detection). Ce protocole permet de détecter si un voisin est inaccessible pour divers raisons. La machine effectuant cette requête a la possibilité d'effacer de ses tables de configurations la machine qui n'est plus accessible. La capture de la Figure 12 montre précisément la façon dont est implémenté ce protocole dans IPv6. Celle-ci est basée sur le schéma réseau de la Figure 11.

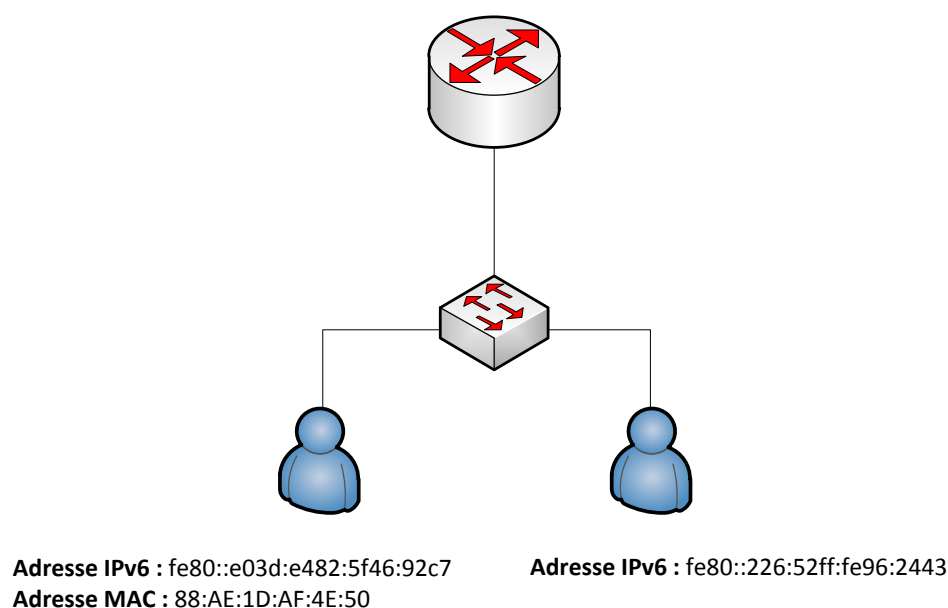


Figure 11 : Schéma du réseau local permettant d'illustrer la fonctionnalité « Neighbor Unreachability Detection » du protocole « Neighbor Discovery ».

fe80::e03d:e482:5f46:92c7	fe80::226:52ff:fe96:2443	ICMPv6 Neighbor solicitation for fe80::226:52ff:fe96:2443 from 88:ae:1d:af:4e:50
fe80::226:52ff:fe96:2443	fe80::e03d:e482:5f46:92c7	ICMPv6 Neighbor advertisement fe80::226:52ff:fe96:2443 (rtr, sol)

Figure 12 : Paquets échangés sur le réseau local de la Figure 11 et illustrant le mécanisme « Neighbor Unreachability Detection ».

La machine source envoie une requête « Neighbor solicitation » à la machine dont elle veut savoir si elle est toujours opérationnelle. Dans cette requête, elle mentionne son adresse MAC dans l'option ICMPv6 « Source link-layer address » (Figure 13).

```

ICMPv6 option (Source link-layer address)
  Type: Source link-layer address (1)
  Length: 8
  Link-layer address: 88:ae:1d:af:4e:50
    
```

Figure 13 : Analyse du paquet « Neighbor solicitation » tiré de la Figure 12.

A la Figure 14, la machine de destination répond par un « Neighbor advertisement » à la machine source ce qui prouve qu'elle est toujours « alive ».

```

Internet Control Message Protocol v6
  Type: 136 (Neighbor advertisement)
  Code: 0
  Checksum: 0x155c [correct]
  Flags: 0xc0000000
  Target: fe80::226:52ff:fe96:2443 (fe80::226:52ff:fe96:2443)

```

Figure 14 : Analyse du paquet « Neighbor advertisement » tiré de la Figure 12.

Si tel n'était pas le cas, l'équipement source effacera la machine de son cache « résolution de voisin » et n'enverra plus de requête « NUD » afin de ne pas générer du trafic inutilement.

### 5.1.3 DAD

La détection d'adresses dupliquées appelée « DAD » (Duplicate Address Detection) est une autre fonctionnalité offerte par NDP. Celle-ci permet, comme son nom l'indique, de détecter si une adresse est déjà utilisée sur le réseau sur lequel veut se connecter une machine. En IPv4, une machine dont l'adresse a été configurée manuellement doit s'assurer que celle-ci est libre grâce au protocole ARP. IPv6 définit cette fonctionnalité au travers du NDP en envoyant tout d'abord une requête « Neighbor solicitation » pour l'adresse IP qu'elle veut s'attribuer comme le montre la Figure 15.

::	ff02::1:ff00:ab	ICMPv6 Neighbor solicitation for 2001:4da0:c00::ab
fe80::ac90:758c:83fc:4c9c	ff02::16	ICMPv6 Multicast Listener Report Message v2
2001:4da0:c00::ab	ff02::1	ICMPv6 Neighbor advertisement 2001:4da0:c00::ab (ovr) is at 88:ae:1d:af:4e:50

Figure 15 : Paquets ICMPv6 représentant la requête « Neighbor solicitation » envoyée par une machine désirant s'assurer que son adresse n'est pas déjà utilisée sur le réseau. Si c'est le cas, un « Neighbor advertisement » est renvoyé par cette dernière.

Ce paquet ICMPv6 contient l'adresse à tester sur le réseau (Figure 16). A noter également que l'adresse d'envoi de ce paquet est l'adresse provisoire de la cible, en l'occurrence « ff02::1:ff00:ab ». Ce type d'adresse provisoire est utilisé uniquement pour envoyer et recevoir des « Neighbor solicitation » et « Neighbor advertisement ».

```

Internet Control Message Protocol v6
  Type: 135 (Neighbor solicitation)
  Code: 0
  Checksum: 0xffb0 [correct]
  Reserved: 0 (Should always be zero)
  Target: 2001:4da0:c00::ab (2001:4da0:c00::ab)

```

Figure 16 : Analyse du paquet « Neighbor solicitation » tiré de la Figure 15.

Une fois la requête envoyée, trois cas différents sont à distinguer :

- Un « Neighbor advertisement » est reçu d'un voisin (Figure 15). Ce paquet indique que l'adresse requise est déjà utilisée par cette machine. La machine ayant envoyé la requête d'origine ne peut donc pas choisir cette adresse.

```

Internet Control Message Protocol v6
  Type: 136 (Neighbor advertisement)
  Code: 0
  Checksum: 0x6d59 [correct]
  Flags: 0x20000000
    0... .. = Not router
    .0.. .. = Not advertised
    ..1. .. = Override
  Target: 2001:4da0:c00::ab (2001:4da0:c00::ab)
  ICMPv6 Option (Target link-layer address)
    Type: Target link-layer address (2)
    Length: 8
    Link-layer address: 88:ae:1d:af:4e:50

```

Figure 17 : Analyse du paquet « Neighbor advertisement » tiré de la Figure 15.

- Un autre « Neighbor solicitation » est reçu en même temps par un autre voisin. Dans ce cas, aucune des deux machines ne peut choisir cette adresse, et elles doivent émettre une nouvelle demande avec une nouvelle adresse.
- Si au bout d'une seconde, la machine ayant émis la requête « DAD » ne reçoit rien, cela signifie que cette adresse est libre et elle peut donc la garder.

Le DAD n'est donc pas une nouvelle fonctionnalité d'IPv6, puisqu'elle était déjà présente en IPv4. Le concept, à savoir qu'une machine envoie une requête sur le réseau pour savoir si l'adresse qu'elle veut prendre est libre ou non, est similaire dans les deux normes, si ce n'est que c'est toujours le protocole NDP qui effectue de contrôle en IPv6.

#### 5.1.4 MTU

En plus des services comme « DHCP » ou « DNS » qui constituent la base de toute connexion à un réseau, un autre élément très important à considérer est la découverte du MTU (Maximum Transmission Unit) sur un lien. Pour rappel, le MTU représente la taille maximum d'un paquet qui peut être transmis par une interface en une seule fois. De ce fait, certains routeurs ont des interfaces dont les MTU sont plus grandes que d'autres, et il est nécessaire d'ajuster la taille des paquets envoyés depuis la source. En IPv4, cet ajustement se faisait par le biais du protocole ICMP. En IPv6, on retrouve le même fonctionnement par le biais du protocole ICMPv6.

En premier lieu, lorsque la machine se connecte au réseau, celle-ci envoie un « Router solicitation » (Figure 18) pour déterminer quel routeur va lui faire office de « gateway ». Le routeur répond ensuite par un « Router advertisement ».

fe80::e03d:e482:5f46:92c7	ff02::2	ICMPv6	Router solicitation from 88:ae:1d:af:4e:50
fe80::aea0:16ff:fe3e:3d7f	ff02::1	ICMPv6	Router advertisement from ac:a0:16:3e:3d:7f

Figure 18 : Paquets ICMPv6 échangés entre un client et un routeur lorsque le client veut connaître sa « gateway ».

C'est dans les options ICMPv6 de ce paquet que le routeur annonce le MTU maximum transmissible sur le lien comme le montre la Figure 19.

```

    ICMPv6 Option (MTU)
      Type: MTU (5)
      Length: 8
      MTU: 1500
  
```

Figure 19 : Analyse du « Routeur advertisement » permettant de montrer la transmission du MTU par les options ICMPv6.

Une fois le MTU connu, la machine envoyant des paquets dont la taille serait plus grande que 1500 octets aura l'obligation de les fragmenter pour que ceux-ci puissent être transmis sur le lien. Si elle ne le fait pas, le paquet ne pourra tout simplement pas être envoyé au destinataire. A noter qu'on retrouve exactement les mêmes concepts qu'en IPv4. La Figure 20 présente les paquets capturés lorsque la commande suivante a été utilisée sur la machine source :

```
ping <adresse_ip_destinataire> -l 2000
```

Cette commande permet d'envoyer une requête ping ayant une longueur de 2000, donc volontairement plus grande que le MTU maximum, afin de voir comment se passe la fragmentation.

IPv6	IPv6 fragment (nxt=ICMPv6 (0x3a) off=0 id=0x8)
ICMPv6	Echo (ping) request id=0x0001, seq=7
IPv6	IPv6 fragment (nxt=ICMPv6 (0x3a) off=0 id=0x20)
ICMPv6	Echo (ping) reply id=0x0001, seq=7
IPv6	IPv6 fragment (nxt=ICMPv6 (0x3a) off=0 id=0xa)
ICMPv6	Echo (ping) request id=0x0001, seq=8
IPv6	IPv6 fragment (nxt=ICMPv6 (0x3a) off=0 id=0x22)
ICMPv6	Echo (ping) reply id=0x0001, seq=8
IPv6	IPv6 fragment (nxt=ICMPv6 (0x3a) off=0 id=0xc)
ICMPv6	Echo (ping) request id=0x0001, seq=9

Figure 20 : Capture des fragments envoyés par la source puisque le MTU fait 1500 et que les données envoyées sont plus grandes.

Le cas précédent illustre donc la découverte du MTU sur le lien sur lequel est connectée la machine. Par contre, si le MTU d'un lien sur lequel n'est pas connectée la machine est inférieure à la taille des paquets que celle-ci veut envoyer, un autre mécanisme doit être mis en œuvre. Dans ce cas, la machine source n'a aucun moyen de savoir au préalable quelle est la taille maximum des paquets qu'elle peut envoyer. Encore une fois, on retrouve le même concept qu'en IPv4. La machine va tout d'abord envoyer son paquet normalement. Une fois ce paquet arrivé au routeur dont le MTU est inférieur, celui-ci va renvoyer un message ICMPv6 « Too big » à la machine source pour lui indiquer le MTU maximum que celui-ci supporte. Ceci est illustré par la Figure 21.

```

IPv6    IPv6 fragment (nxt=ICMPv6 (0x3a) off=0 id=0x10)
ICMPv6  Echo (ping) request id=0x0001, seq=11
ICMPv6  Too big (Unknown (0x00))
IPv6    IPv6 fragment (nxt=ICMPv6 (0x3a) off=0 id=0x12)
ICMPv6  Echo (ping) request id=0x0001, seq=12
IPv6    IPv6 fragment (nxt=ICMPv6 (0x3a) off=0 id=0x28)
ICMPv6  Echo (ping) reply id=0x0001, seq=12
IPv6    IPv6 fragment (nxt=ICMPv6 (0x3a) off=0 id=0x14)
ICMPv6  Echo (ping) request id=0x0001, seq=13

```

Figure 21 : Paquets échangés permettant d'illustrer le paquet « Too big » envoyé par un routeur dont le MTU est inférieure à la taille du paquet envoyé.

Ce paquet « Too big » contient le nouveau MTU que la machine source doit prendre en considération. Tout comme dans le premier cas, celle-ci va donc fragmenter ses paquets pour que ceux-ci ne dépassent pas 1300 octets, qui est le MTU transmis par le routeur (Figure 22).

```

Internet Control Message Protocol v6
  Type: 2 (Too big)
  Code: 0 (Unknown)
  Checksum: 0x50fd [correct]
  MTU: 1300

```

Figure 22 : Analyse du paquet « Too big » de la Figure 21 permettant d'indiquer à la machine source qu'un lien sur le chemin entre elle et le destinataire ne supporte pas la taille des paquets envoyés.

Un autre cas de figure peut être rencontré si deux routeurs (ou plus) ne supporte pas la taille du paquet envoyé, et que le deuxième a un MTU plus faible que le premier. Dans ce cas, le premier routeur va indiquer que le paquet est trop grand par le message ICMPv6 « Too big ». Une fois que la machine source le reçoit, celle-ci va fragmenter ses paquets pour correspondre à ce nouveau MTU, et les renvoyer. Ceux-ci vont ensuite arriver au deuxième routeur dont le MTU est plus faible que l'autre, et va également renvoyer le message « Too big » pour que la machine source s'adapte. Les paquets échangés sont illustrés par la Figure 23. Bien sûr, si le MTU du premier routeur est inférieur à celui du deuxième, un seul message « Too big » sera renvoyé puisque la machine s'adaptera au premier MTU reçu qui sera plus faible que l'autre.

```

IPv6 fragment (nxt=ICMPv6 (0x3a) off=0 id=0x18)
Echo (ping) request id=0x0001, seq=15
Too big (Unknown (0x00))
IPv6 fragment (nxt=ICMPv6 (0x3a) off=0 id=0x1a)
Echo (ping) request id=0x0001, seq=16
Too big (Unknown (0x00))
IPv6 fragment (nxt=ICMPv6 (0x3a) off=0 id=0x1c)
Echo (ping) request id=0x0001, seq=17

```

Figure 23 : Paquets échangés lorsque deux routeurs sur le chemin que va prendre un paquet ont un MTU inférieur à ce qu'envoie la machine source.

Le mécanisme de MTU présent en IPv6 est donc en tout point similaire à ce qui existe actuellement en IPv4. La machine envoie son paquet, et le routeur dont le MTU est plus petit l'informe, par l'intermédiaire du protocole ICMPv6 (ICMP en IPv4) qu'il est nécessaire de fragmenter le paquet. Celle-ci peut donc le fragmenter et le renvoyer sur le lien. La seule différence à tenir compte est qu'en IPv4, les routeurs ont la possibilité de fragmenter le paquet si le bit DF (Don't Fragment) vaut 0. Ce bit n'existe plus en IPv6 et c'est forcément la machine source qui va fragmenter le paquet.

## 5.2 DHCPv6

Le protocole DHCP est un élément indispensable pour toute machine voulant communiquer avec d'autres sur un réseau. En effet, sans lui, les ordinateurs n'auraient aucun moyen de se procurer une adresse, si ce n'est de se l'être vu affiliée manuellement par un administrateur, ce qui est évidemment à proscrire lorsque le réseau contient beaucoup d'ordinateurs. Ces derniers vont donc, une fois connectés, contacter un serveur DHCP afin de lui demander une adresse disponible. A noter que ce serveur peut très bien être un routeur, modem, ou une machine serveur. En IPv4, il existe donc une seule façon de se procurer une adresse auprès d'un serveur DHCP. En IPv6 par contre, la chose diffère passablement puisqu'il existe plusieurs modes de configuration d'adresse pour le client.

Ce chapitre décrit donc précisément les différents mécanismes mis en œuvre en IPv6 pour qu'un client se procure une adresse auprès du serveur. Chaque mode est présenté séparément pour une meilleure compréhension de la problématique du protocole DHCPv6. Si cela est nécessaire, des scénarios basés sur différents schémas de réseau sont mis en œuvre, afin de pouvoir distinguer le cas où le client se trouve sur le même sous-réseau que le serveur du cas où les deux ne s'y trouvent pas. Enfin, les paquets les plus importants sont détaillés précisément pour illustrer le fonctionnement de ce protocole.

Les différents modes de configuration d'adresses présents en IPv6 sont donc les suivants : (si on ne tient pas compte de la configuration manuelle possible dans les deux) :

- Statefull
- Auto-configuration
- Stateless

Chaque mode implique ses spécificités dans la prise d'adresse d'un client et est détaillé précisément dans les sections qui suivent.

### 5.2.1 Mode « Statefull »

Le mode statefull hérite du fonctionnement du mode standard sous IPv4. De manière résumée, dans l'ancienne norme, un client a besoin d'une adresse pour pouvoir communiquer sur le réseau local, il émet une requête au serveur DHCP et celui-ci lui répond avec une adresse. En IPv6 mode statefull, cela se passe exactement la même chose.

#### 5.2.1.1 Global

Le premier scénario mis en œuvre pour illustrer ce mode se réfère au schéma de la Figure 24, toujours basé sur le prototype principal :

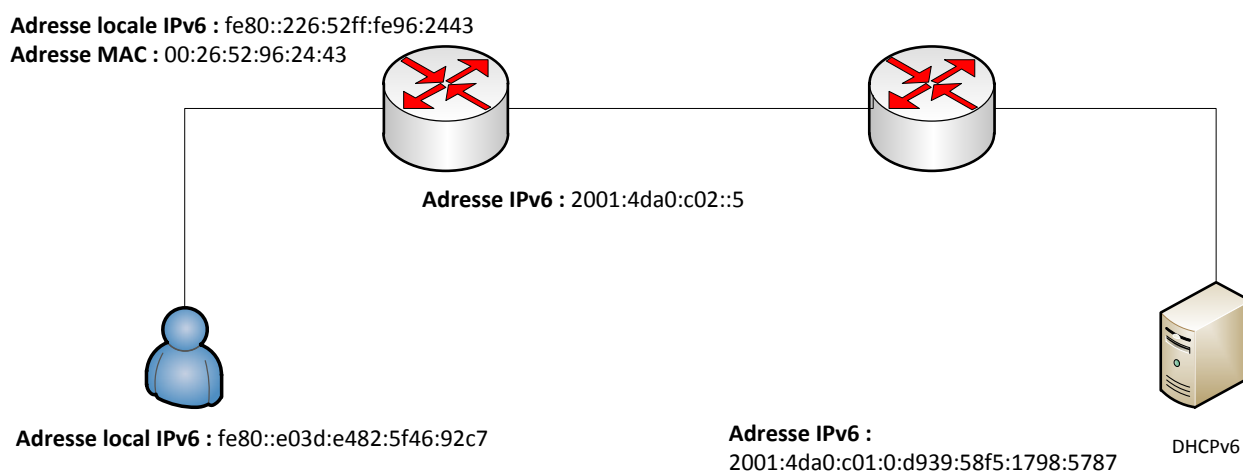


Figure 24 : Schéma du réseau permettant d'illustrer le mode « Statefull » du DHCP lors du client et le serveur sur deux sous-réseaux différents.

On constate que le serveur DHCP n'est pas présent sur le même sous-réseau que le client. Ceci va bien évidemment impliquer un comportement différent par rapport au cas où les deux sont situés sur le même sous-réseau, qui sera détaillé dans la sous-section qui suit.

Le poste client, une fois branché au réseau, commence par envoyer une requête « Router solicitation » à l'adresse multicast ff02::2 afin de contacter le routeur qui lui fera ensuite office de passerelle par défaut. La Figure 25 présente les différents paquets ICMPv6 échangés entre le client et le routeur qui va lui faire office de passerelle par défaut.

fe80::e03d:e482:5f46:92c7	ff02::2	ICMPv6 Router solicitation from 88:ae:1d:af:4e:50
fe80::e03d:e482:5f46:92c7	ff02::16	ICMPv6 Multicast Listener Report Message v2
fe80::226:52ff:fe96:2443	ff02::1	ICMPv6 Router advertisement from 00:26:52:96:24:43

Figure 25 : Paquets ICMPv6 envoyés par un client lui permettant de contacter le routeur lui faisant office de passerelle.

Le routeur répond à cette requête par un « Router advertisement » qui comporte également un flag servant au

client afin qu'il sache comment récupérer son adresse. Dans le cas d'un statefull, le flag binaire est positionné sur « Managed » (Figure 26), ce qui implique que le client doit contacter un DHCP pour récupérer son adresse.

```

Flags: 0x80
1... .... = Managed
.0.. .... = Not other
..0. .... = Not Home Agent
...0 0... = Router preference: Medium
.... .0.. = Not Proxied
    
```

Figure 26 : Analyse du « Flag » obtenu lors d'une configuration « statefull ».

**Remarque :** Pour plus d'informations, se référer au travail de M. Tissot expliquant en détail les différents flags.

Une fois que le client reçoit le paquet lui indiquant que le « flag » contient la valeur « Managed », celui-ci va donc contacter le DHCP. Les 6 paquets de la Figure 27 vont donc être échangés entre le client et le serveur durant la phase d'acquisition d'adresse par le client.

fe80::e03d:e482:5f46:92c7	ff02::1:2	DHCPv6 Confirm XID:
fe80::226:52ff:fe96:2443	fe80::e03d:e482:5f46:92c7	DHCPv6 Reply XID: 0x
fe80::e03d:e482:5f46:92c7	ff02::1:2	DHCPv6 Solicit XID:
fe80::226:52ff:fe96:2443	fe80::e03d:e482:5f46:92c7	DHCPv6 Advertise XID
fe80::e03d:e482:5f46:92c7	ff02::1:2	DHCPv6 Request XID:
fe80::226:52ff:fe96:2443	fe80::e03d:e482:5f46:92c7	DHCPv6 Reply XID: 0x

Figure 27 : Paquets échangés en configuration « statefull » entre un client et le DHCP qu'il contacte afin d'obtenir son adresse IPv6.

Les deux premiers paquets sont capturés uniquement du fait que c'est un PC Windows 7 qui a effectué les tests. En effet ces derniers enregistrent les derniers paramètres IP qu'ils se sont vus attribuer et lorsqu'ils se reconnectent à un réseau, commencent par demander si ceux-ci sont toujours correctes. Si on analyse plus en détail le paquet « Confirm » (Figure 28), on remarque que le poste demande si l'adresse 2001:4da0:c01:0:9d59:bd3b:7e0d:98ad est toujours valable. Remarquons que ceci est totalement similaire à celle effectuée en IPv4.

```

Identity Association for Non-temporary Address
Option: Identity Association for Non-temporary Address (3)
Length: 40
Value: 1188ae1d0000000000000000000000005001820014da0c010000...
IAID: 1188ae1d
T1: 0
T2: 0
IA Address: 2001:4da0:c01:0:9d59:bd3b:7e0d:98ad
    
```

Figure 28 : Analyse du paquet « Confirm » de la Figure 27 contenant l'adresse dont le client veut vérifier la validité.

Le DHCP répond par la négative grâce au « Reply » et le client commence à partir de ce moment-là la procédure de demande d'adresse auprès du serveur. Celle-ci se passe de la manière suivante : Le client envoie un « Solicit » sur l'adresse multicast ff02::1:2 qui est l'adresse multicast « d'écoute » des serveurs DHCP. Le serveur répond au



client par un « Advertise » et lui envoie son adresse IP comme le montre la Figure 29.

```

[+] IA Address: 2001:4da0:c00:0:5d85:13de:1953:cd2b
    Option: IA Address (5)
    Length: 24
    Value: 20014da0c0000005d8513de1953cd2b000151800002a300
    IPv6 address: 2001:4da0:c00:0:5d85:13de:1953:cd2b
    Preferred lifetime: 86400
    valid lifetime: 172800
  
```

Figure 29 : Analyse du paquet « Advertise » de la Figure 27 renvoyé par le serveur et contenant la nouvelle adresse IP du client.

A partir de cette étape, le client possède une adresse IP. Il va finalement demander les différents paramètres dont il a besoin, le plus courant étant l'adresse du serveur DNS, ou dans le cas de la voix sur IP, les adresses des serveurs SIP par exemple. Ceci se fait donc au moyen du paquet « Request » et le serveur répond avec le paquet « Reply » (Figure 27) qui contient les informations souhaitées.

```

[+] Domain Search List
[+] DNS recursive name server
    Option: DNS recursive name server (23)
    Length: 16
    Value: 20014da0c010000d93958f517985787
    DNS servers address: 2001:4da0:c01:0:d939:58f5:1798:5787
[+] Fully Qualified Domain Name
  
```

Figure 30 : Analyse du paquet « Reply » de la Figure 27 renvoyé par le serveur et indiquant au client les informations supplémentaires souhaitées.

Une fois ceci analysé, il est nécessaire d'expliquer la présence d'adresses de source et de destination de liaison locale, puisque il a été dit que le serveur ne se trouve pas sur le même sous-réseau. Cela vient du fait que lorsque le serveur se trouve dans un autre sous réseau que le client, ce dernier ne peut évidemment pas le contacter directement puisqu'il n'a pas d'adresse globale. C'est donc le routeur faisant office de passerelle par défaut du client qui va jouer le rôle de relai entre le client et le DHCP. La Figure 31 illustre les paquets reçus par le serveur dans cette configuration :

2001:4da0:c02::5	2001:4da0:c01:0:d939:58f5:1798:5787	DHCPv6 Relay-forw L:
2001:4da0:c01:0:d939:58f5:1798:5787	2001:4da0:c02::5	DHCPv6 Relay-reply L:
2001:4da0:c02::5	2001:4da0:c01:0:d939:58f5:1798:5787	DHCPv6 Relay-forw L:
2001:4da0:c01:0:d939:58f5:1798:5787	2001:4da0:c02::5	DHCPv6 Relay-reply L:
2001:4da0:c02::5	2001:4da0:c01:0:d939:58f5:1798:5787	DHCPv6 Relay-forw L:
2001:4da0:c01:0:d939:58f5:1798:5787	2001:4da0:c02::5	DHCPv6 Relay-reply L:

Figure 31 : Paquets capturés sur le serveur selon le schéma de la Figure 24 et illustrant le principe de relai lorsque le client voulant contacter le DHCP ne se trouve pas sur le même sous-réseau que celui-ci.

On constate que les adresses de cette capture sont cette fois-ci des adresses globales, plus précisément les adresses du routeur et du serveur. Les paquets reçus par le routeur sont donc encapsulés pour pouvoir atteindre la destination globale du serveur DHCP. Si on prend par exemple le paquet « Solicit » envoyé par le client et qu'on

l'analyse, on remarque clairement le contenu de ce paquet « intégré » au paquet dont les adresses sources sont globales :

```

Relay Message
  Option: Relay Message (9)
  Length: 145
  Value: 02dac8950002000e000100004d777ee0005056a300770001...
  DHCPv6
    Message type: Advertise (2)
    Transaction ID: 0xdac895
    Server Identifier: 000100004d777ee0005056a30077
    Client Identifier: 0001000114fc3b2d88ae1daf4e50
    Identity Association for Non-temporary Address
      Option: Identity Association for Non-temporary Address (3)
      Length: 40
      Value: 1188ae1d0000a8c000010e000005001820014da00c000000...
      IAID: 1188ae1d
      T1: 43200
      T2: 69120
    IA Address: 2001:4da0:c00:0:5d85:13de:1953:cd2b
  
```

Figure 32 : Analyse du paquet « Relay-reply » de la Figure 31 qui encapsule le paquet « Advertise » lorsque le client ne se trouve pas sur le même sous-réseau que le serveur.

Ainsi, le routeur peut transmettre tous les paquets du client vers le serveur, et le serveur lui répondre directement. Le routeur n'a plus qu'à désencapsuler les réponses du serveur pour les transmettre au client.

Après l'analyse de ce mode en configuration « globale », on remarque que son fonctionnement est très similaire à celui observé en IPv4, où le routeur fait office de relai entre le client et le serveur lorsque ceux-ci sont situés sur des sous-réseaux différents. La partie suivante décrit le mode statefull lorsque les deux sont cette fois-ci situés dans le même réseau.

### 5.2.1.2 Local

Bien que cela soit plutôt rare, on peut imaginer que les clients soient connectés au même sous-réseau que le serveur DHCP. Les explications qui vont suivre vont donc être basées sur le schéma de la Figure 33.

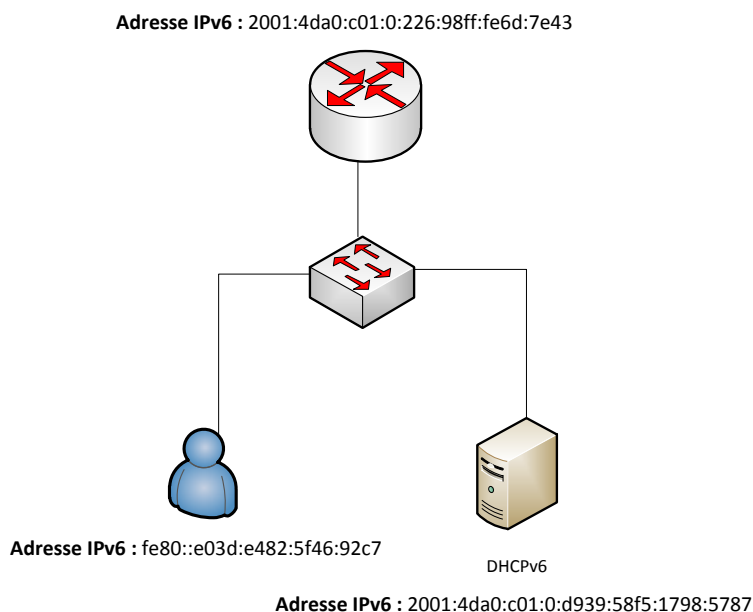


Figure 33 : Schéma du réseau local permettant d'illustrer le mode « statefull » du DHCP lorsque le client et le serveur se trouvent sur le même sous-réseau.

Une fois le client branché, la Figure 34 présente les différents paquets DHCPv6 échangés avec le serveur.

fe80::e03d:e482:5f46:92c7	ff02::1:2	DHCPv6 Solicit XID: 0x288433 CID: 0001000114fc3b2d88ae1daf4e50
2001:4da0:c01:0:d939:58f5:1798:5787	fe80::e03d:e482:5f46:92c7	DHCPv6 Advertise XID: 0x288433 CID: 0001000114fc3b2d88ae1daf4e50
fe80::e03d:e482:5f46:92c7	ff02::1:2	DHCPv6 Request XID: 0x288433 CID: 0001000114fc3b2d88ae1daf4e50 I
2001:4da0:c01:0:d939:58f5:1798:5787	fe80::e03d:e482:5f46:92c7	DHCPv6 Reply XID: 0x288433 CID: 0001000114fc3b2d88ae1daf4e50 IAA

Figure 34 : Paquets échangés sur le réseau de la Figure 33 entre le client et le serveur DHCP lors d'une demande d'adresse.

On voit logiquement que les mêmes paquets sont échangés par rapport au statefull global. La différence notable entre les deux configurations réside dans le fait que lors du statefull global, comme le client s'adressait au serveur par l'intermédiaire du routeur sur un lien local, les adresses étaient donc uniquement des adresses locales. A l'inverse, lorsque le client est situé dans le même sous-réseau, on remarque qu'il envoie toujours ses requêtes à l'adresse multicast, mais que c'est bel et bien le serveur qui lui répond directement, avec comme adresse source son adresse globale.

Un autre point important de cette configuration est que l'option « dhcp-relay » a été volontairement désactivée sur le routeur faisant office de passerelle pour le client, ceci du fait que le routeur n'a pas besoin de relayer les messages entre le client et le serveur, ceux-ci se trouvant sur le même sous-réseau. Si cette option n'avait pas été désactivée, et qu'une demande de renouvellement d'adresse est effectuée à l'aide de la commande « ipconfig /renew » la capture obtenue serait la suivante :

fe80::f8cd:246b:de87:ac75	ff02::1:2	DHCPv6 Release XID: 0
2001:4da0:c01:0:d939:58f5:1798:5787	fe80::f8cd:246b:de87:ac75	DHCPv6 Reply XID: 0
2001:4da0:c01:0:226:98ff:fe6d:7e43	2001:4da0:c01:0:d939:58f5:1798:5787	DHCPv6 Relay-forw L
2001:4da0:c01:0:d939:58f5:1798:5787	2001:4da0:c01:0:226:98ff:fe6d:7e43	DHCPv6 Relay-reply
fe80::f8cd:246b:de87:ac75	ff02::1:2	DHCPv6 solicit XID: 0
2001:4da0:c01:0:d939:58f5:1798:5787	fe80::f8cd:246b:de87:ac75	DHCPv6 Advertise XID: 0
2001:4da0:c01:0:226:98ff:fe6d:7e43	2001:4da0:c01:0:d939:58f5:1798:5787	DHCPv6 Relay-forw L
2001:4da0:c01:0:d939:58f5:1798:5787	2001:4da0:c01:0:226:98ff:fe6d:7e43	DHCPv6 Relay-reply
fe80::f8cd:246b:de87:ac75	ff02::1:2	DHCPv6 Request XID: 0
2001:4da0:c01:0:d939:58f5:1798:5787	fe80::f8cd:246b:de87:ac75	DHCPv6 Reply XID: 0
2001:4da0:c01:0:226:98ff:fe6d:7e43	2001:4da0:c01:0:d939:58f5:1798:5787	DHCPv6 Relay-forw L
2001:4da0:c01:0:d939:58f5:1798:5787	2001:4da0:c01:0:226:98ff:fe6d:7e43	DHCPv6 Relay-reply
fe80::f8cd:246b:de87:ac75	ff02::1:2	DHCPv6 Renew XID: 0
2001:4da0:c01:0:d939:58f5:1798:5787	fe80::f8cd:246b:de87:ac75	DHCPv6 Reply XID: 0
2001:4da0:c01:0:226:98ff:fe6d:7e43	2001:4da0:c01:0:d939:58f5:1798:5787	DHCPv6 Relay-forw L
2001:4da0:c01:0:d939:58f5:1798:5787	2001:4da0:c01:0:226:98ff:fe6d:7e43	DHCPv6 Relay-reply

Figure 35 : Paquets capturés sur le serveur selon le schéma de la Figure 33 lorsque la fonctionnalité de relai est activée sur le routeur.

**Remarque :** Il est tout à fait normal que l'adresse IPv6 locale de la machine faisant la requête d'adresse ait changé puisque la commande `ipconfig /renew` la force à reconstruire son adresse.

Il est inutile de détailler chaque paquet de cette capture, puisqu'aucun changement n'est effectué par rapport à une demande d'adresse standard. La chose importante à constater par contre, est que des paquets « Relay-form » et « Relay-reply » apparaissent lorsque le routeur fait office de relai. Les paquets sont donc reçus « à double » par le client et le serveur puisque pour chaque paquet transmis entre les deux, le routeur doit les relayer au destinataire. Ceci explique les nombreux paquets capturés de ce type. A noter que ceci n'influence en aucun cas la prise d'adresse du client auprès du serveur.

Le mode statefull est donc en tout point similaire à ce qui se fait actuellement en IPv4, avec une gestion des adresses centralisée sur un serveur. Cela permet de pouvoir configurer quelles adresses doivent être données aux clients, et lesquelles ne doivent pas l'être. Les possibilités qu'offrent les DHCP sont multiples et dépendent du système utilisé. La présentation des différentes étapes de la configuration d'un serveur DHCP sous Windows Server 2008 est présentée dans la partie « Configuration DHCPv6 » en annexe.

### 5.2.2 Auto-configuration

Le deuxième « mode » d'acquisition d'adresse pour un client est l'auto-configuration. Celui-ci n'implique aucun serveur DHCP puisque le client va configurer son adresse lui-même grâce au préfixe réseau qu'il reçoit du routeur. Comme il n'y a pas de concept de « local » ou « global », le schéma de référence est donc celui de la Figure 36.

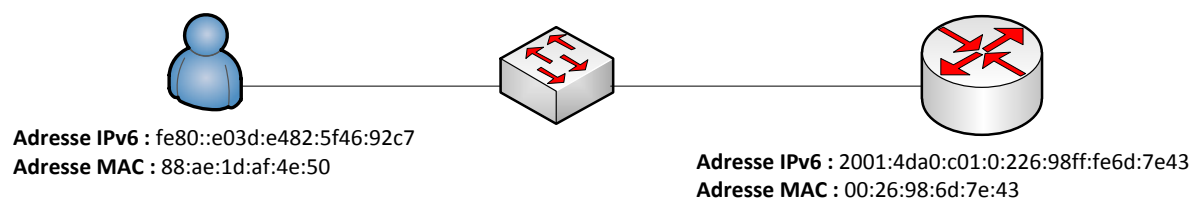


Figure 36 : Schéma du réseau local permettant d'illustrer le mode « auto-configuration ».

En analysant les paquets ICMPv6 échangés (Figure 37), on observe le « Router solicitation » envoyé par le client pour solliciter son routeur, et l'« advertisement » renvoyé par ce dernier :

fe80::e03d:e482:5f46:92c7	ff02::2	ICMPv6 Router solicitation from 88:ae:1d:af:4e:50
fe80::e03d:e482:5f46:92c7	ff02::16	ICMPv6 Multicast Listener Report Message v2
fe80::226:98ff:fe6d:7e43	ff02::1	ICMPv6 Router advertisement from 00:26:98:6d:7e:43

Figure 37 : Paquets ICMPv6 échangés sur le schéma de la Figure 36 et permettant à un client de connaître sa passerelle par défaut.

Plus en détail, on remarque dans le « Router advertisement » (Figure 38) tout d'abord que le préfixe et sa longueur sont transmis au client, contrairement au mode statefull où le client ne reçoit pas sa longueur. La conséquence logique de ceci est qu'en statefull, un ping va devoir forcément passer par le routeur puisque le client n'a aucun moyen de savoir si la destination qu'il veut atteindre se situe sur le même sous-réseau que lui ou non. Par contre, en auto-configuration (tout comme en stateless d'ailleurs), le ping peut directement être adressé sur le lien local.

```

ICMPv6 option (Prefix information)
  Type: Prefix information (3)
  Length: 32
  Prefix Length: 64
  Flags: 0xc0
    1... .... = on-link flag(L): Set
    .1.. .... = Autonomous address-configuration flag(A): Set
    ..00 0000 = Reserved: 0
  Valid lifetime: 2592000
  Preferred lifetime: 604800
  Reserved
  Prefix: 2001:4da0:c01::
  
```

Figure 38 : Analyse de l'option « Prefix information » envoyée par le routeur au client par le biais du paquet « Router advertisement » selon le schéma de la Figure 36.

La deuxième chose que l'on remarque est la différence de flag qui renseigne le client que l'adresse doit être auto-

configurée par lui-même. A ce stade, on peut se demander comment cela est pratiquement utilisable, puisque dans ce cas de configuration, le client n'a aucun moyen d'obtenir l'adresse du DNS vers qui il doit s'adresser pour résoudre les noms ou les adresses IP. Le mécanisme mis en place dans ce cas est le suivant : chaque ordinateur Windows dispose d'adresses DNS « par défaut » comme illustré à la Figure 39.

```

Serveurs DNS. . . . . : fec0:0:0:ffff::1%1
                      : fec0:0:0:ffff::2%1
                      : fec0:0:0:ffff::3%1

```

Figure 39 : Illustration des différentes adresses DNS que les machines Windows ont par défaut dans le cas où aucune adresse de DNS n'est transmise au client.

Il est ainsi possible aux clients de faire leurs requêtes, en configurant correctement les routeurs pour que ceux-ci puissent router ces adresses vers le serveur DNS.

Ce mode est donc un concept nouveau dans le protocole DHCPv6 par rapport à l'ancienne norme. Il permet une configuration de chaque interface de chaque ordinateur sans pour autant avoir une gestion centralisée par un serveur DHCP. Elle peut être particulièrement utile dans le cas où l'on branche des machines à un réseau simple ne contenant pas de DHCP. Elles seront donc capables de communiquer entre elles directement, sans aucune configuration nécessaire de la part de l'administrateur. A noter également que comme expliqué dans la partie « DAD », les machines vont bien évidemment tester si l'adresse qu'elles se sont attribuées est unique sur le réseau, afin d'éviter tout conflit.

### 5.2.3 Mode « Stateless »

Le mode stateless, tout comme le mode « auto-configuration » présenté dans la partie précédente, est un tout nouveau mode de configuration d'adresse introduit par la norme IPv6. Celui-ci permet à un client de configurer lui-même son adresse au moyen du préfixe réseau envoyé par le routeur et d'aller **chercher sur le serveur DHCP uniquement les paramètres dont il a besoin**. Tout comme pour le mode statefull, deux cas sont à distinguer selon que le client est situé dans le même sous-réseau que le serveur ou non.

### 5.2.3.1 Global

Le schéma pour ce mode illustré par la Figure 40, est le même que pour le mode statefull.

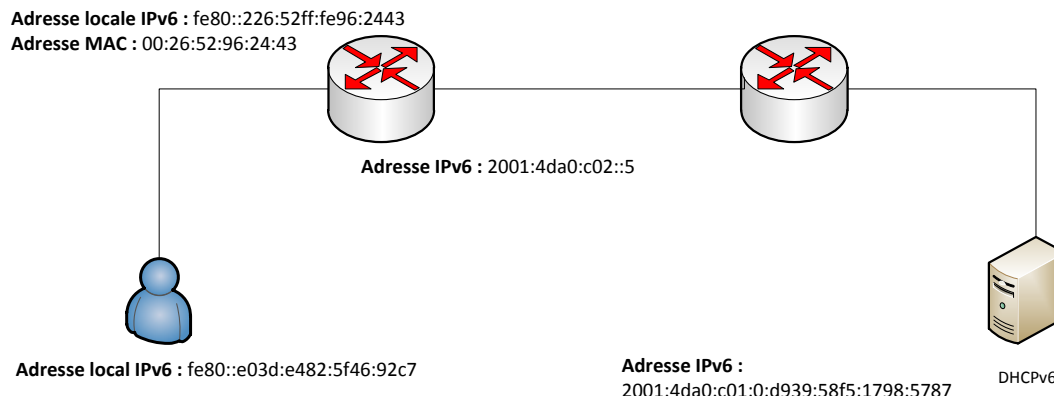


Figure 40 : Schéma du réseau local permettant d'illustrer le mode « stateless » du DHCP lorsque le client et le serveur se trouvent sur deux sous-réseaux différents.

Tout comme en mode statefull, une fois le poste branché au réseau, celui-ci envoie une requête « Router solicitation » pour découvrir sa passerelle par défaut (Figure 41).

fe80::e03d:e482:5f46:92c7	ff02::2	ICMPv6 Router solicitation from 88:ae:1d:af:4e:50
fe80::e03d:e482:5f46:92c7	ff02::16	ICMPv6 Multicast Listener Report Message v2
fe80::226:52ff:fe96:2443	ff02::1	ICMPv6 Router advertisement from 00:26:52:96:24:43

Figure 41 : Paquets ICMPv6 échangés sur le schéma de la Figure 40 et permettant à un client de connaître sa passerelle par défaut.

Le routeur répond ensuite par un « Router advertisement » (Figure 42) dans lequel est présent le « flag » (Figure 43) qui permet de déterminer la façon dont le client va acquérir son adresse.

```

ICMPv6 Option (Source link-layer address)
  Type: Source link-layer address (1)
  Length: 8
  Link-layer address: 88:ae:1d:af:4e:50
    
```

Figure 42 : Analyse de l'option « Source link-layer address » envoyée par le routeur au client par le biais du paquet « Router advertisement » selon le schéma de la Figure 40.

Contrairement en mode statefull, cette fois-ci ce « flag » est en mode « Other », ce qui indique au client qu'il doit construire son adresse lui-même mais qu'il peut aller chercher des paramètres sur le DHCP :

```

Flags: 0x40
0... .... = Not managed
.1.. .... = Other
..0. .... = Not Home Agent
...0 0... = Router preference: Medium
.... .0.. = Not Proxied
  
```

Figure 43 : Analyse du « Flag » obtenu lors d'une configuration « stateless ».

Une autre différence est que cette fois-ci, le routeur envoie également le préfixe du réseau afin que le client puisse se constituer son adresse lui-même (Figure 44). Dans notre cas, le préfixe envoyé est `2001:4da0:c00::`:

```

ICMPv6 Option (Prefix information)
Type: Prefix information (3)
Length: 32
Prefix Length: 64
Flags: 0xc0
valid lifetime: 2592000
Preferred lifetime: 604800
Reserved
Prefix: 2001:4da0:c00::
  
```

Figure 44 : Analyse de l'option « Prefix information » envoyée par le routeur au client par le biais du paquet « Router advertisement » selon le schéma de la Figure 40.

A ce stade, le client peut, par les mécanismes présentés au début du rapport, se créer sa propre adresse IP sans pour autant avoir contacté un DHCP, ce qui est une grande nouveauté de la norme. Il a également la possibilité d'aller chercher les informations dont il a besoin sur celui-ci.

2001:4da0:c02::5	2001:4da0:c01:0:d939:58f5:1798:5787	DHCPv6 Relay-forw L: 2
2001:4da0:c01:0:d939:58f5:1798:5787	2001:4da0:c02::5	DHCPv6 Relay-reply L:

Figure 45 : Paquets DHCPv6 « Relay » échangés selon le schéma de la Figure 40 entre le routeur faisant office de relai entre le client et le serveur DHCP, et le DHCP lui-même.

Comme le client ne connaît pas l'adresse du serveur DHCP, le même mécanisme de relai est également employé dans ce cas, illustré par la Figure 45. Ainsi, c'est le routeur qui fait office d'intermédiaire entre le client et le serveur pour que le premier puisse demander les paramètres au deuxième. Un exemple des paramètres demandés est donné grâce à la figure ci-dessous.



```

[-] Relay Message
  Option: Relay Message (9)
  Length: 58
  Value: 0b6baed00008000200c80001000e0001000114fc3b2d88ae...
[-] DHCPv6
  Message type: Information-request (11)
  Transaction ID: 0x6baed0
  [+ Elapsed time
  [+ Client Identifier: 0001000114fc3b2d88ae1daf4e50
  [+ Vendor Class
  [-] Option Request
    Option: Option Request (6)
    Length: 8
    Value: 0018001700110020
    Requested Option code: Domain Search List (24)
    Requested Option code: DNS recursive name server (23)
    Requested Option code: Vendor-specific Information (17)
    Requested Option code: Lifetime (32)
  
```

Figure 46 : Analyse du paquet « Relay-reply » de la Figure 40 qui encapsule le paquet « Information-request » lorsque le client ne se trouve pas sur le même sous-réseau que le serveur.

Dans ce cas, on remarque que quatre options sont fournies au client. Celles-ci sont paramétrables sur le serveur au moyen des options présentées dans la partie « Scope options ».

### 5.2.3.2 Local

En stateless local, la configuration de référence est représentée par la Figure 47.

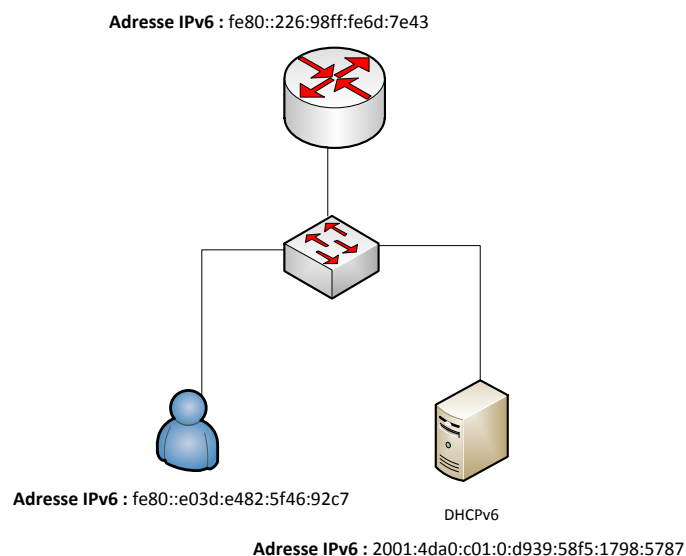


Figure 47 : Schéma du réseau local permettant d'illustrer le mode « stateless » du DHCP lorsque le client et le serveur se trouvent sur le même sous-réseau.

Le schéma est donc le même que celui de la Figure 33 sur lequel se base l'explication du mode « statefull ».

La Figure 48 montre donc qu'il n'y a pas de différence liée au fait que le serveur DHCP se trouve sur le même sous-réseau que le client.

fe80::e03d:e482:5f46:92c7	ff02::2	ICMPv6 Router solicitation from 88:ae:1d:af:4e:50
fe80::e03d:e482:5f46:92c7	ff02::16	ICMPv6 Multicast Listener Report Message v2
fe80::226:98ff:fe6d:7e43	ff02::1	ICMPv6 Router advertisement from 00:26:98:6d:7e:43

Figure 48 : Paquets ICMPv6 échangés sur le schéma de la Figure 48 et permettant à un client de connaître sa passerelle par défaut.

On remarque toujours la découverte du routeur par le client et l'annonce de celui-ci au client, avec la transmission du préfixe de réseau pour qu'il puisse se construire sa propre adresse (Figure 49).

```

- ICMPv6 Option (Prefix information)
  Type: Prefix information (3)
  Length: 32
  Prefix Length: 64
  + Flags: 0xc0
    valid lifetime: 2592000
    Preferred lifetime: 604800
    Reserved
    Prefix: 2001:4da0:c01::

```

Figure 49 : Analyse de l'option « Prefix information » envoyée par le routeur au client par le biais du paquet « Router Advertisement » selon le schéma de la Figure 47.

Par contre, étant donné que le routeur est obligé de relayer les paquets entre le serveur et le client (puisque'il n'a aucun moyen de savoir que les demandes du client sont adressées à un serveur local), le client va recevoir à chaque fois deux « DHCPv6 Reply » (Figure 50), un venant du serveur et un autre relayé par le routeur.

fe80::e03d:e482:5f46:92c7	ff02::1:2	DHCPv6 Information-request
2001:4da0:c01:0:d939:58f5:1798:5787	fe80::e03d:e482:5f46:92c7	DHCPv6 Reply XID: 0x5ddb2b
fe80::226:98ff:fe6d:7e43	fe80::e03d:e482:5f46:92c7	DHCPv6 Reply XID: 0x5ddb2b

Figure 50 : Paquets reçus par le client lorsque la fonctionnalité de relais n'est pas désactivée sur le routeur et que le serveur se trouve dans le même réseau que le client selon le schéma de la Figure 47.

Ceci n'a aucune incidence sur l'acquisition des paramètres par le client.

Le mode stateless est donc un mode IPv6 très utile puisqu'il permet à un client, tout en s'auto-configurant son adresse, d'aller chercher des paramètres sur le serveur. A la différence du mode « statefull », la gestion du DHCP est nettement plus simple, puisqu'il n'y a pas besoin de prévoir des étendues d'adresses par sous-réseau, et la configuration de celui-ci s'en trouve facilitée. L'administrateur réseau a donc le choix entre plusieurs modes de configuration DHCP en fonction du réseau qu'il gère. Ceci représente un avantage certain sur le fonctionnement du protocole DHCP en IPv4, où le choix se résumait à savoir si un serveur était requis ou non.

#### 5.2.4 Scope options

Les scopes options du serveur DHCPv6 de Microsoft Server 2008 sont présentées dans cette partie. Un constat important à faire est que celles-ci sont nettement moins nombreuses que les scopes options d'IPv4. Voici brève présentation de chacune d'entre elles :

##### **00021 SIP Server Domain Name List**

Cette option permet de définir une liste des noms de domaine des serveurs proxy SIP que le client va utiliser pour faire de la VOIP

##### **00022 SIP Server IPv6 Address List**

Toujours dans le cadre de la VOIP, ce paramètre permet de spécifier une liste d'adresses IPv6 des proxys SIP utilisés par le client. Notons que les adresses doivent être indiquées dans l'ordre de préférence des serveurs.

##### **00023 DNS Recursive Name Server IPv6 Address**

Cette option permet de fournir une liste d'adresses IPv6 RDNSS au client afin que celui-ci sache à qui envoyer ses requêtes DNS. Tout comme pour l'option 22, l'ordre de préférence des serveurs DNS doit être respecté.

##### **00024 Domain Search List**

Cette option correspond à l'option « 00015 DNS Domain Name » d'IPv4 et permet de définir une liste de nom de domaine à utiliser pour résoudre les noms d'hôtes lors de résolution DNS.

##### **00027 NIS IPv6 Address List**

L'option 00027 est similaire à l'option 00022 à la différence que ce sont les adresses IPv6 des serveurs NIS (Network Information Services) qui sont mentionnés, toujours dans l'ordre de préférence

##### **00028 NIS + IPv6 Address List**

Même option que la 00027, mais indique les adresses IPv6 des serveurs NIS fonctionnant en version 2.

##### **00029 NIS Domain List**

Permet de spécifier la liste des noms de domaine des serveurs NIS.

##### **00030 NIS + Domain Name List**

Même option que la 00029 mais pour les serveurs NIS fonctionnant en version 2.

## **00031SNTP Servers IPv6 Address List**

Cette option permet de définir les adresses IPv6 des serveurs SNTP qui sont à la disposition du client pour qu'il puisse se synchroniser au niveau du temps.

## **00032 Information Refresh Time**

Cette option permet de définir le temps de rafraichissement des informations reçues par les clients du serveur DHCP.

## 5.3 DNS

Le protocole DNS est un autre protocole de couche 7 extrêmement utilisé. Pour rappel, celui-ci permet de travailler avec des noms de domaine plutôt qu'avec des adresses IP. Il est en effet nettement plus simple pour un être humain de se souvenir de l'adresse `www.google.com` plutôt que de l'adresse IP `209.85.148.99`. Ce protocole définit donc la façon dont les clients vont interroger le serveur DNS afin de transcrire une adresse en un nom de domaine ou inversement.

En IPv6, le protocole DNS n'implique pas de réels changements par rapport à IPv4. Dans ce cas, il est préférable de parler d'extension du protocole, puisqu'en effet les paquets transmis sont les mêmes. Il est par contre essentiel dans la mesure où il est très fastidieux de travailler avec des adresses IPv6 complètes, dans le cas par exemple d'administration de routeurs ou de connexion quelconque.

**Remarque :** Pour la configuration d'un serveur DNS sur Windows Server 2008, veuillez se référer en annexe dans la partie « Configuration DNS ».

### 5.3.1 Nommage direct et inverse

Pour rappel, le service DNS définit deux types de nommage : le nommage direct et le nommage inverse. Le premier est utilisé pour récupérer l'adresse IP d'un nom et, à l'opposé, le second permet une résolution de l'adresse vers le nom. En IPv4, le nommage direct s'effectue grâce au type d'enregistrement « A » et de manière analogue, IPv6 définit des types « AAAA » (prononcé « quad A »). La capture de la Figure 52 nous montre différents types d'enregistrement direct en IPv6 :

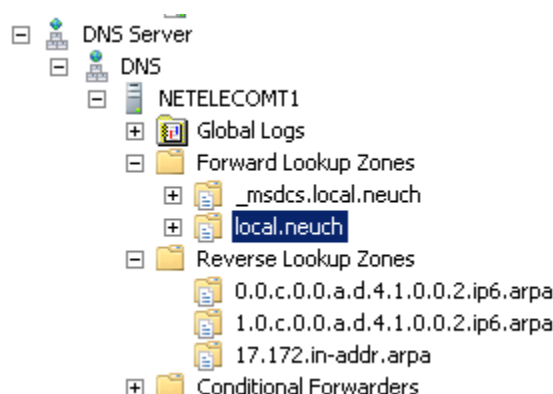


Figure 51 : Zone directe « local.neuch » contenant les enregistrements figurant sur la Figure 52.

netelecomt1	Host (A)	172.17.128.2	static
netelecomt1	IPv6 Host (AAAA)	2001:4da0:0c01:0000:d939:58f5:1798:5788	static
netelecomt1	IPv6 Host (AAAA)	2001:4da0:0c01:0000:d939:58f5:1798:5787	static
pcRouge	Host (A)	172.17.1.7	18.05.2011 14:00:00
pcRouge	IPv6 Host (AAAA)	2001:4da0:0c00:0001:f8cd:246b:de87:ac75	18.05.2011 14:00:00
rt1841	IPv6 Host (AAAA)	2001:4da0:0c02:0001:0000:0000:0000:0003	static
rt2811	IPv6 Host (AAAA)	2001:4da0:0c02:0001:0000:0000:0000:0002	static
STEVE	IPv6 Host (AAAA)	2001:4da0:0c00:0000:150c:9cbd:0605:ec81	18.05.2011 13:00:00
sw1	IPv6 Host (AAAA)	2001:4da0:0c02:0001:0000:0000:0000:0000	static
sw2	IPv6 Host (AAAA)	2001:4da0:0c02:0001:0000:0000:0000:0001	static
sw3	IPv6 Host (AAAA)	2001:4da0:0c00:0001:0000:0000:0000:0001	static

Figure 52 : Enregistrements directs des différents éléments connectés au domaine « local.neuch ».

Comme une machine peut avoir plusieurs adresses IP globales, le serveur DNS contient autant d'enregistrement de type AAAA qu'il y a d'adresses IPv6 pour une machine définie. Ainsi, lorsqu'une requête de type AAAA est effectuée, la réponse contiendra toutes les adresses correspondantes au nom demandé. On remarque également que les adresses peuvent être enregistrées statiquement ou dynamiquement, tout comme en IPv4. La problématique de l'enregistrement dynamique est détaillée dans la partie suivante. Le nommage inverse est par contre en tout point similaire à celui effectué en IPv4, et se nomme toujours PTR. Si on regarde les enregistrements de ce type, voici ce que l'on obtient :

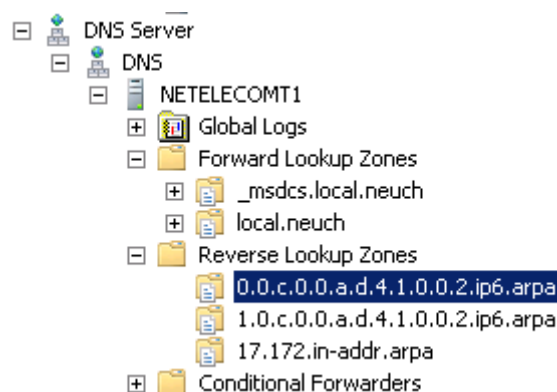


Figure 53 : Zone inverse « local.neuch » contenant les enregistrements figurant sur la Figure 54.

Les zones inverses sont toujours écrites avec le format « *in-addr.arpa* » où l'adresse IP du réseau est écrite à l'envers. Le contenu de chaque zone contient, comme dans l'ancienne norme, des pointeurs sur les différents noms en fonction des adresses :

2001:4da0:0c00:0000:ac90:7...	Pointer (PTR)	julien-pc.local.neuch.	11.05.2011 11:00:00
2001:4da0:0c00:0001:ac90:7...	Pointer (PTR)	julien-pc.local.neuch.	18.05.2011 10:00:00
2001:4da0:0c00:0001:f95b:7...	Pointer (PTR)	pcrouge.local.neuch.	11.05.2011 15:00:00

Figure 54 : Enregistrements indirectes des différents éléments connectés au domaine « local.neuch ».

### 5.3.2 Enregistrement dynamique

Il paraît évident que les administrateurs n'aient pas à entrer chaque nom de machine dans le DNS pour la résolution de ceux-ci mais que cela se fasse automatiquement par le biais d'enregistrements dynamiques. Tout comme en IPv4, cela est possible en version 6, grâce aux requêtes de type « Dynamic update SOA ». Avant toute chose, il est préférable qu'un Active Directory soit installé sur Windows Server 2008 afin que les enregistrements dynamiques des clients puissent se faire, le service DNS étant fortement lié au service de l'AD dans la plupart des cas. Le paramètre principal à définir pour chaque zone est la sécurité des mises à jour dynamiques. En effet, il existe trois types de sécurité par rapport à celles-ci qui sont illustrées à la Figure 55.

- **None** : Aucune mise à jour dynamique n'est autorisée.
- **Nonsecure and secure** : Aucune sécurité n'est activée, n'importe quelle machine peut s'inscrire dans le DNS
- **Secure only** : Seuls les clients membre d'une zone de l'AD sont autorisés à s'inscrire dans le DNS.

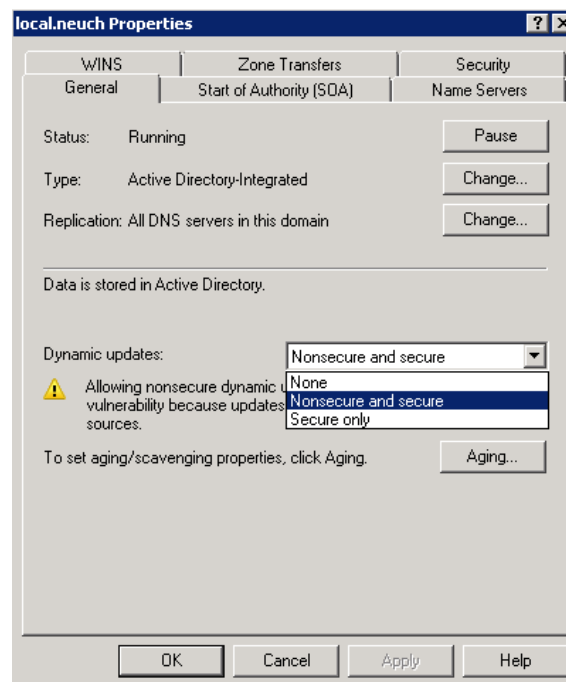


Figure 55 : Capture illustrant les différents types de sécurité possibles des mises à jour DNS.

L'enregistrement dynamique peut s'effectuer aussi via le DHCP. Pour ceci, il suffit de configurer chaque zone du DHCP avec les paramètres voulus comme le montre la Figure 56.

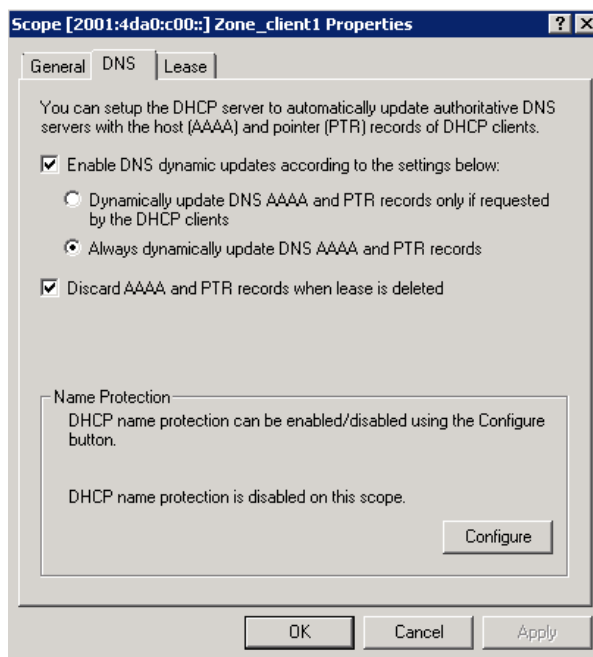


Figure 56 : Capture illustrant les différentes configurations possibles paramétrables sur chaque zone du DHCP pour les mises à jour dynamiques du DNS.

Le premier paramètre permet de définir si les clients récupérant une adresse auprès du serveur DHCP doivent être enregistrés sur le serveur DNS ou non. Les deux suivants permettent soit de choisir d'enregistrer le client si celui-ci le demande, soit par défaut d'enregistrer tous les clients. Enfin, la dernière option permet d'effacer les enregistrements AAAA et PTR du DNS quand le bail de l'adresse a expiré. Certaines configurations sont également paramétrables sur le client, dans les paramètres IP avancé de l'onglet DNS de la carte réseau (Figure 57).



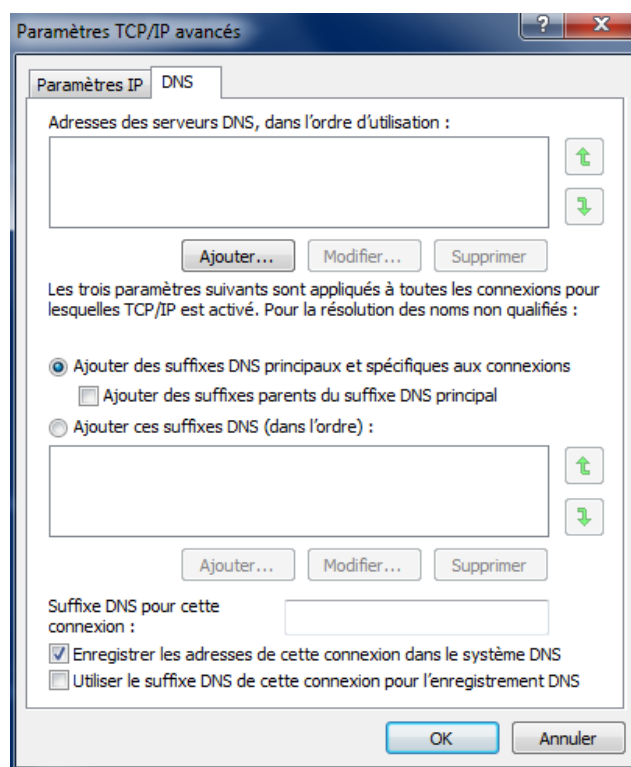


Figure 57 : Capture montrant un exemple de configuration de la carte réseau à adopter afin que l'hôte s'enregistre dynamiquement dans le système DNS.

Le seul qui doit absolument être sélectionné est « Enregistrer les adresses de cette connexion dans le système DNS ». Cette configuration est donc similaire à ce que l'on retrouve en IPv4. Une fois cette option sélectionnée et le client branché au réseau, voici ce qui est capturé :

2001:4da0:c01:0:79d2:502:21af:340	2001:4da0:c01 DNS	Dynamic update SOA local.neuch
2001:4da0:c01:0:d939:58f5:1798:5787	2001:4da0:c01 DNS	Dynamic update response CNAME
2001:4da0:c01:0:79d2:502:21af:340	2001:4da0:c01 DNS	Standard query SOA c.9.c.4.c.f.3.8.c.8.5.7.0.9.c.a.0.
2001:4da0:c01:0:d939:58f5:1798:5787	2001:4da0:c01 DNS	Standard query response, No such name
2001:4da0:c01:0:79d2:502:21af:340	2001:4da0:c01 DNS	Dynamic update SOA 1.0.c.0.0.a.d.4.1.0.0.2.ip6.arpa
2001:4da0:c01:0:d939:58f5:1798:5788	2001:4da0:c01 DNS	Dynamic update response CNAME

Figure 58 : Paquets échangés entre le client et le serveur DNS lorsque celui-ci émet une requête de type « SOA » afin d'enregistrer son nom dynamiquement.

Le client, une fois connecté, émet une requête de type SOA vers le nom de son domaine. Cette requête contient les éléments présentés à la Figure 59.

```

- Updates
  - Julien-PC.local.neuch: type AAAA, class ANY
  - Julien-PC.local.neuch: type A, class ANY
  - Julien-PC.local.neuch: type AAAA, class IN, addr 2001:4da0:c01:0:ac90:758c:83fc:4c9c
    Name: Julien-PC.local.neuch
    Type: AAAA (IPv6 address)
    Class: IN (0x0001)
    Time to live: 20 minutes
    Data length: 16
    Addr: 2001:4da0:c01:0:ac90:758c:83fc:4c9c

```

Figure 59 : Analyse du paquet DNS « Dynamic Updates SOA local.neuch » de la Figure 58 et permettant au client d'enregistrer dynamiquement son nom dans la zone directe du DNS.

Ceci indique au serveur que le client veut inscrire son nom dans la zone directe du DNS avec son adresse IP. Le serveur répond ensuite pour valider la requête grâce au « Dynamic update response ». Ensuite, le client tente de s'inscrire dans la zone inverse grâce à une deuxième requête SOA dans lequel on trouve les informations de la Figure 60.

```

- Updates
  - c.9.c.4.c.f.3.8.c.8.5.7.0.9.c.a.0.0.0.0.1.0.c.0.0.a.d.4.1.0.0.2.ip6.arpa: type PTR, class ANY
    Name: c.9.c.4.c.f.3.8.c.8.5.7.0.9.c.a.0.0.0.0.1.0.c.0.0.a.d.4.1.0.0.2.ip6.arpa
    Type: PTR (Domain name pointer)
    Class: ANY (0x00ff)
    Time to live: 0 time
    Data length: 0

```

Figure 60 : Analyse du paquet DNS « Dynamic Updates [zone indirecte] » de la Figure 58 et permettant au client d'enregistrer dynamiquement son nom dans la zone indirecte du DNS.

Une fois les deux requêtes effectuées, le nom est accessible dans le DNS. La sélection du stack IP si IPv4 et IPv6 sont activés est expliquée dans la partie suivante. Si la sécurité est activée sur le serveur DNS et que donc celui-ci refuse les mises à jour dynamiques, voici ce qui est capturé (Figure 61).

172.17.128.6	172.17.128.2	DNS	Dynamic update SOA local.neuch
172.17.128.2	172.17.128.6	DNS	Dynamic update response, Refused CNAME
172.17.128.6	172.17.128.2	DNS	Dynamic update SOA local.neuch
172.17.128.2	172.17.128.6	DNS	Dynamic update response, Refused CNAME
172.17.128.2	172.17.128.6	DNS	Standard query response TKEY
172.17.128.6	172.17.128.2	DNS	Dynamic update SOA local.neuch
172.17.128.2	172.17.128.6	DNS	Dynamic update response, Refused CNAME
2001:4da0:c01:0:2008:43b9:9791:9933	2001:4da0:c01:0:d939:58f5:1798:5787	DNS	Standard query SOA STEVE.local.neuch
2001:4da0:c01:0:d939:58f5:1798:5787	2001:4da0:c01:0:2008:43b9:9791:9933	DNS	Standard query response
172.17.128.6	172.17.128.2	DNS	Dynamic update SOA local.neuch
172.17.128.2	172.17.128.6	DNS	Dynamic update response, Refused CNAME
172.17.128.6	172.17.128.2	DNS	Dynamic update SOA local.neuch
172.17.128.2	172.17.128.6	DNS	Dynamic update response, Refused CNAME
172.17.128.2	172.17.128.6	DNS	Standard query response TKEY
172.17.128.6	172.17.128.2	DNS	Dynamic update SOA local.neuch
172.17.128.2	172.17.128.6	DNS	Dynamic update response, Refused CNAME

Figure 61 : Paquets échangés lorsque la mise à jour dynamique du DNS est refusée par le serveur. Ceci intervient en fonction du type de sécurité de mises à jour choisies comme illustré à la Figure 55.

La capture nous montre clairement que les requêtes sont refusées par les réponses « Dynamic update response, Refused CNAME ». Ceci explique le fait que la sécurité sur le DNS doit être paramétrée correctement, faute de

quoi les clients ne pourront pas s'y inscrire. A noter que sur la capture, les requêtes sont faites en IPv4, car les deux stacks IP sont activés sur le client. Ceci n'a aucune incidence par rapport aux options de sécurités DNS.

### 5.3.3 DNS dual-stack

Le dual-stack représente un concept plutôt vague dans le sens où il n'y a pas de norme qui régit la priorité du stack à adopter. Ainsi, certaines applications vont préférer IPv4 et donc choisir cette norme là où d'autres choisirait IPv6 ou 6to4. Pour ce qui se rapporte au DNS, si les deux stacks sont activés sur la machine, c'est IPv4 qui va être choisi pour effectuer l'enregistrement dynamique comme on peut le voir sur la capture ci-dessous.

172.17.128.6	172.17.128.2	DNS	Dynamic update SOA local.neuch
172.17.128.2	172.17.128.6	DNS	Dynamic update response CNAME
2001:4da0:c01:0:2008:43b9:9791:9933	2001:4da0:c01:0:d939:58f5:1798:5787	DNS	Standard query SOA 6.128.17.172.in-addr.arpa
2001:4da0:c01:0:d939:58f5:1798:5787	2001:4da0:c01:0:2008:43b9:9791:9933	DNS	Standard query response
172.17.128.6	172.17.128.2	DNS	Dynamic update SOA 17.172.in-addr.arpa
172.17.128.2	172.17.128.6	DNS	Dynamic update response CNAME
2001:4da0:c01:0:2008:43b9:9791:9933	2001:4da0:c01:0:d939:58f5:1798:5787	DNS	Standard query SOA STEVE.local.neuch
2001:4da0:c01:0:d939:58f5:1798:5787	2001:4da0:c01:0:2008:43b9:9791:9933	DNS	Standard query response
172.17.128.6	172.17.128.2	DNS	Dynamic update SOA local.neuch
172.17.128.2	172.17.128.6	DNS	Dynamic update response CNAME
2001:4da0:c01:0:2008:43b9:9791:9933	2001:4da0:c01:0:d939:58f5:1798:5787	DNS	Standard query SOA 7.c.2.9.6.4.f.5.2.8.4.e.d.3.0.e.0
2001:4da0:c01:0:d939:58f5:1798:5787	2001:4da0:c01:0:2008:43b9:9791:9933	DNS	Standard query response, No such name
172.17.128.6	172.17.128.2	DNS	Dynamic update SOA 1.0.c.0.0.a.d.4.1.0.0.2.ip6.arpa

Figure 62 : Paquets échangés lors d'une mise à jour dynamique du DNS en « dual-stack ». Cette capture permet de voir la sélection du stack IPv4.

On remarque bien que les paquets « update SOA » sont adressés en IPv4 au serveur. La Figure 63 montre clairement la mise à jour de l'enregistrement de la machine « STEVE.local.neuch » avec les deux normes : L'enregistrement A pour IPv4 ainsi que l'enregistrement AAAA pour IPv6.

```

Zone
├─ local.neuch: type SOA, class IN
Prerequisites
├─ STEVE.local.neuch: type CNAME, class NONE
Updates
├─ STEVE.local.neuch: type AAAA, class ANY
├─ STEVE.local.neuch: type A, class ANY
├─ STEVE.local.neuch: type AAAA, class IN, addr 2001:4da0:c01:0:e03d:e482:5f46:92c7
└─ STEVE.local.neuch: type A, class IN, addr 172.17.128.6

```

Figure 63 : Analyse du paquet « Dynamic update SOA local.neuch » de la Figure 62 et permettant au client d'enregistrer ses adresses IPv4 et IPv6.

## 6 Sécurité

Comme dans toutes applications ou normes informatiques, la sécurité représente un point sensible qu'il est primordial de prendre en compte lors d'un développement, que ce soit celui d'un petit programme ou d'une norme mondiale. Actuellement, les attaques contre les systèmes informatiques sont de plus en plus nombreuses et proviennent dans la plupart des cas de « pirates » jeunes et inexpérimentés. Ceci implique qu'elles sont pour la plupart simples à effectuer et qu'elles profitent de vulnérabilités connues. Il suffit donc à un non-expert de savoir utiliser un moteur de recherche pour qu'il parvienne à lancer des attaques qui peuvent faire des dégâts considérables en termes de confidentialité, d'intégrité et de disponibilité.

Cette partie a donc pour but de sensibiliser sur les attaques possibles du protocole IPv6 dans un réseau local. Il est présenté la façon de mettre en œuvre différents types d'attaques bien connues en IPv4 que sont le « Man in the middle », le « Deny of Service » ou encore le « Scanning ». Il est également important de noter que cette partie constitue une simple introduction au sujet, tant celui-ci est vaste et complexe. Elle ne constitue en aucun cas une référence absolue, puisqu'il s'agirait d'un travail complet en la matière.

Afin de pouvoir tester différentes failles de sécurité spécifiques au protocole NDP d'IPv6, il a été mis en place le schéma de la Figure 64 où une personne malveillante attaquerait une victime du même sous réseau.

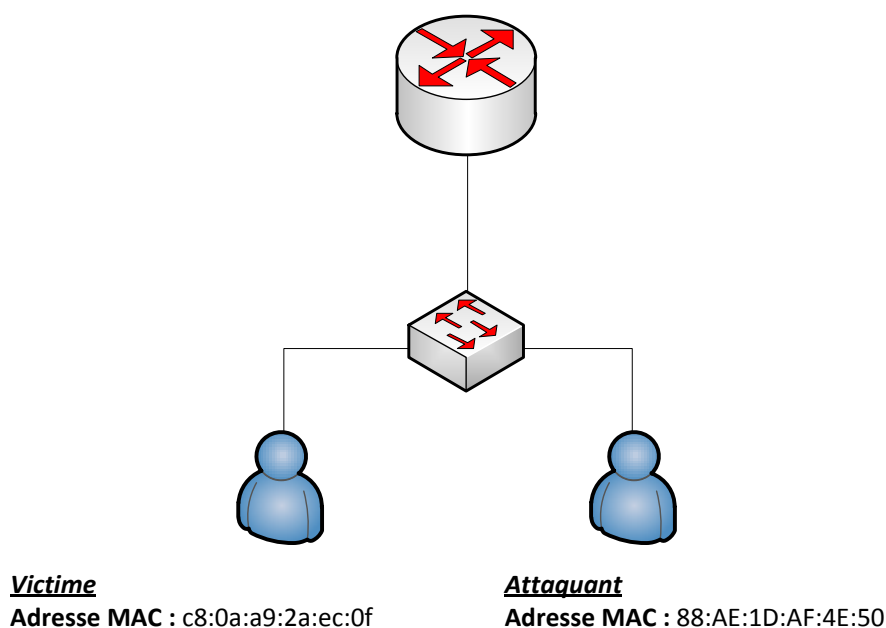


Figure 64 : Schéma du réseau local permettant d'illustrer différentes attaques possible sur le protocole IPv6 lorsque l'attaquant et la victime se trouvent dans le même sous-réseau.

Ce schéma fait donc office de référence pour les différentes parties traitant de la sécurité qui vont suivre.

## 6.1 Toolkit

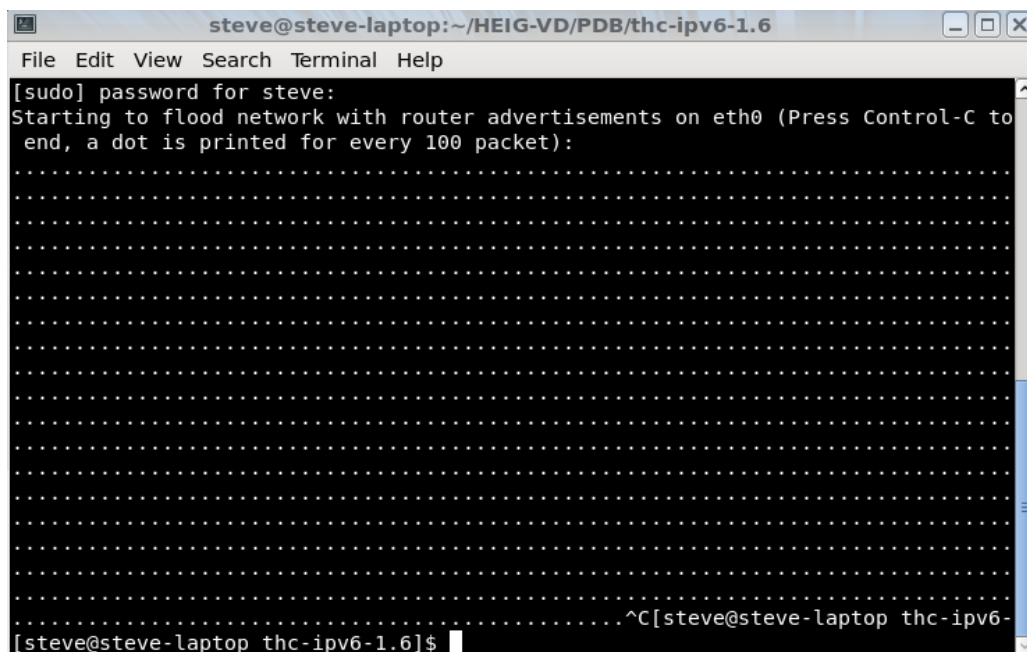
Les attaques mises en œuvre dans cette partie peuvent être réalisées avec le toolkit téléchargeable à l'adresse suivante : <http://www.thc.org/thc-ipv6/>

Ce toolkit est sous licence GPL. Il est donc totalement légal de s'en servir à un usage personnel. Celui-ci inclut tout une série de petits scripts permettant de réaliser des attaques locales sur le protocole IPv6. Toutes les attaques n'ont toutefois pas été testées pour des raisons évidentes de temps. Seules les attaques les plus utiles à connaître pour un attaquant mais surtout pour le défenseur du réseau sont présentées.

**Remarque :** Le toolkit est utilisable exclusivement sur un système GNU/Linux.

## 6.2 Windows Denial of Service

La première attaque présentée est celle d'un déni de service sur un poste Windows 7. Celle-ci consiste à « flooder » le réseau en envoyant une quantité énorme de « Router advertisement » comme montré à la Figure 65, où chaque point dans la console correspond à 100 paquets de ce type envoyés.



**Figure 65 : Capture montrant l'utilisation du script "flood\_router6" appartenant au toolkit « thc-ipv6-1.6 ».**

Ainsi, il y a la possibilité pour quelqu'un de bloquer totalement une machine tournant sous Windows 7 en générant des « Router advertisement » en quantité suffisante. Le résultat est immédiat puisque même la souris ne peut plus bouger. La Figure 66 montre plus en détail ce qui est envoyé par la machine de la personne

malveillante.

fe80::218:b1ff:fec2:7a2a	ff02::1	ICMPv6	Router advertisement from 00:18:b1:c2:7a:2a
fe80::218:9cff:febd:f4fe	ff02::1	ICMPv6	Router advertisement from 00:18:9c:bd:f4:fe
fe80::218:45ff:fedf:94eb	ff02::1	ICMPv6	Router advertisement from 00:18:45:df:94:eb
fe80::218:dcff:fe5a:4a56	ff02::1	ICMPv6	Router advertisement from 00:18:dc:5a:4a:56
fe80::218:4fff:feff:e643	ff02::1	ICMPv6	Router advertisement from 00:18:4f:ff:e6:43
fe80::218:f8ff:fe7a:958c	ff02::1	ICMPv6	Router advertisement from 00:18:f8:7a:95:8c
fe80::218:ff:fe43:bf4a	ff02::1	ICMPv6	Router advertisement from 00:18:00:43:bf:4a
fe80::218:61ff:fec8:7047	ff02::1	ICMPv6	Router advertisement from 00:18:61:c8:70:47

Figure 66 : Paquets ICMPv6 représentant un « flooding » de « Router advertisement » afin de saturer la RAM et le processeur d'une machine Windows 7.

On remarque aisément les différents « Router advertisement » envoyés aléatoirement par la machine attaquante. Plus précisément, cette attaque exploite deux particularités. La première est que pour chaque nouvelle adresse de routeur que la machine reçoit, cette dernière va mettre à jour sa table de routage. La seconde est illustrée par le champ « flags » de la Figure 67, qui constitue véritablement le point central de cette attaque.

```

Flags: 0xe0
 1... .... = On-link flag(L): set
 .1.. .... = Autonomous address-configuration flag(A): set
 ..10 0000 = Reserved: 32
 valid lifetime: infinity
 Preferred lifetime: infinity
 Reserved
 Prefix: 2a01:4342:db97:9242::

```

Figure 67 : Analyse d'un des paquets ICMPv6 « Router advertisement » de la Figure 66 permettant de montrer que la valeur du champ « Flags » indique à la machine de destination qu'il doit s'auto-configurer son adresse en fonction du préfixe réseau qu'il reçoit.

Comme on peut le constater, ce champ est initialisé à « auto-configuration », ce qui signifie que la machine va s'auto-configurer une nouvelle adresse pour chaque nouveau préfixe reçu. Etant donné que ceux-ci sont générés aléatoirement à raison d'environ 7000/s, ceci a pour effet de consommer un maximum de la puissance du processeur et de la mémoire RAM, d'où un blocage total de l'ordinateur.

Les impacts de cette vulnérabilité diffèrent d'un système à l'autre. Comme mentionné, les machines Windows 7 se retrouvent saturées. Il n'existe cependant aucun correctif à ce jour proposé par Microsoft pour résoudre ce problème. La seule façon d'éviter que la machine soit vulnérable est de désactiver par la commande suivante la découverte des routeurs (ou alors de désactiver la pile IPv6 de la carte réseau) :

```
netsh interface ipv6 set interface "Local Area Connection" routerdiscovery=disabled
```

Dans le cadre de ce travail, seul le système d'exploitation Windows 7 a été testé. Cependant, il semble que d'autres systèmes, comme FreeBSD ou d'anciens noyaux Linux ainsi que les routeurs Cisco soient impactés. Pour

plus d'informations, se référer au lien suivant :

[http://www.mh-sec.de/downloads/mh-RA\\_flooding\\_CVE-2010-multiple.txt](http://www.mh-sec.de/downloads/mh-RA_flooding_CVE-2010-multiple.txt)

### 6.3 Man in the Middle

L'attaque Man in the Middle est une des attaques les plus courantes dans un réseau local car elle est très pratique pour un attaquant puisque celle-ci lui permet de faire en sorte que tout le trafic en provenance d'une machine vers sa « gateway » passe par lui. La Figure 68 schématise clairement cette attaque assez simple à mettre en œuvre en IPv4.

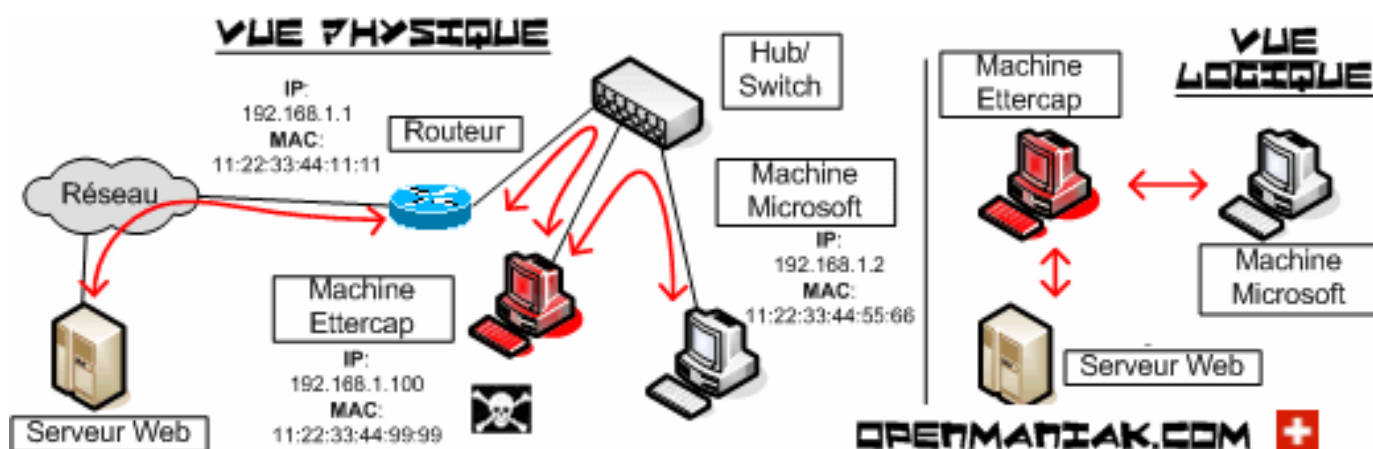


Figure 68 : Illustration du principe de base d'une attaque Man in the middle effectuée en IPv4.

(Source : <http://openmaniak.com>)

Sans entrer dans les détails, cette attaque exploite le protocole ARP, qui permet de faire une translation d'adresse MAC vers une adresse IP dans le cadre de la norme IPv4. Ainsi, lorsqu'une machine va demander à qui appartient l'adresse MAC du routeur, c'est la machine attaquante qui va répondre en court-circuitant la réponse de la passerelle. La victime va donc croire qu'elle s'adresse à la passerelle alors que c'est bel et bien par la machine malicieuse que les paquets vont passer. Sur la figure ci-dessus est présentée une vue physique où on voit que la machine (sur laquelle est installé le logiciel « Ettercap » qui permet de réaliser des attaques de ce type en IPv4) ayant l'adresse MAC 11:22:33:44:99:99 intercepte les paquets en provenance de l'adresse MAC 11:22:33:44:55:66. Ces paquets sont normalement destinés à l'adresse 11:22:33:44:11:11. Ensuite, en activant l'« IP forwarding » sur la machine malveillante, les paquets sont redirigés vers la « default gateway » ce qui fait que la victime ne se rend compte de rien, comme la vue logique de la Figure 68 le montre.

En IPv6, les mécanismes sont exactement les mêmes. La seule différence est qu'au lieu de « spoofer » les paquets ARP, il s'agit de « spoofer » les « Neighbor advertisement » afin que la machine victime croie que l'IP de sa



« default gateway » soit en correspondance avec l'adresse MAC de la machine de l'attaquant. La Figure 69 montre l'utilisation du script « parasite6 » afin de pouvoir se placer entre la victime et le routeur selon le schéma de la Figure 64.

```

steve@steve-laptop: ~/HEIG-VD/PDB/thc-ipv6-1.6
File Edit View Search Terminal Help
[steve@steve-laptop thc-ipv6-1.6]$ sudo ./parasite6 eth0
Remember to enable routing (ip forwarding), you will denial service otherwise!
Started ICMP6 Neighbor Solicitation Interceptor (Press Control-C to end) ...
Spoofed packet to 2001:4444:5555::9 as 2001:4444:5555::1
Spoofed packet to 2001:4444:5555::9 as 2001:4444:5555::1

```

Figure 69 : Capture montrant l'utilisation du script "parasite6" appartenant au toolkit « thc-ipv6-1.6 ».

Une fois la commande lancée, le programme affiche tous les paquets « spoofés ». Sur la Figure 69, on voit deux paquets destinés à l'adresse 2001:4444:5555::1 en provenance de l'adresse 2001:4444:5555::9 qui ont été capturés sur l'interface « eth0 ». En observant les différents paquets transitant (Figure 70), on constate un « Neighbor solicitation » de la part de la victime qui vérifie si le routeur est toujours « alive » (voir NUD). Ensuite, on voit le « Neighbor advertisement » renvoyé par le pirate et qui indique que l'adresse MAC 88:ae:1d:af:4e:50 correspond à l'IP 2001:4444:5555::1. Il est évident que cette attaque se joue sur le fait que le paquet renvoyé par le pirate arrive avant le paquet véridique.

2001:4444:5555::9	2001:4444:5555::1	ICMPv6	Neighbor solicitation for 2001:4444:5555::1 from c8:0a:a9:2a:ec:0f
2001:4444:5555::1	2001:4444:5555::9	ICMPv6	Neighbor advertisement 2001:4444:5555::1 (sol, ovr) is at 88:ae:1d:af:4e:50

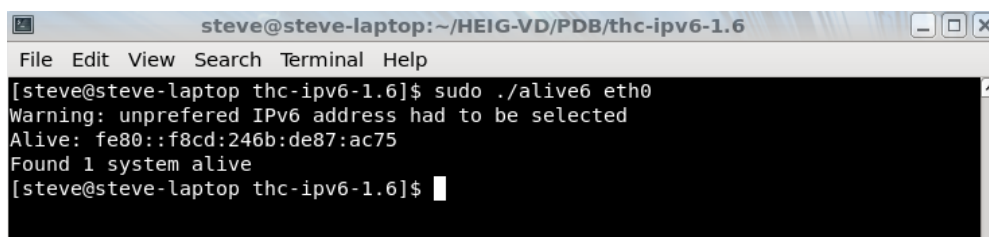
Figure 70 : Paquets ICMPv6 représentant le paquet « Neighbor advertisement » permettant à une personne malveillante de court-circuiter la réponse véridique et de se situer en situation de « Man in the middle ».

Cette attaque démontre le fait que bien qu'on pourrait penser qu'étant donné l'inexistence du protocole ARP dans la nouvelle norme, une attaque de type « Man in The Middle » serait inapplicable mais il n'en est rien. Le protocole NDP y est tout aussi vulnérable comme expliqué ci-dessus. Il est donc très important de pouvoir prendre les mesures adéquates contre ce genre d'attaques très courantes, car très faciles à mettre en œuvre. Celles-ci sont présentées dans les parties « DHCP snooping » et « Secure Neighbor Discovery ».



## 6.4 Scanning

En plus des deux attaques présentées ci-dessus et s'appuyant sur les failles du NDP d'IPv6, un attaquant a également la possibilité de s'informer de tous les systèmes présents sur un réseau local. L'outil « alive6 » du toolkit « thc-ipv6-1.6 » permet de faire un scanning du réseau local en utilisant la commande « sudo ./alive6 eth0 » (Figure 71).



```
steve@steve-laptop: ~/HEIG-VD/PDB/thc-ipv6-1.6
File Edit View Search Terminal Help
[steve@steve-laptop thc-ipv6-1.6]$ sudo ./alive6 eth0
Warning: unprefered IPv6 address had to be selected
Alive: fe80::f8cd:246b:de87:ac75
Found 1 system alive
[steve@steve-laptop thc-ipv6-1.6]$
```

Figure 71 : Capture montrant l'utilisation du script "alive6" appartenant au toolkit « thc-ipv6-1.6 ».

Une personne malveillante utilisant ce script peut aisément connaître tous les systèmes du réseau, ce qui est une information plus que précieuse pour une attaque ultérieure. Bien que cela reste une information basique pour le pirate, le scanning constitue le premier point d'entrée dans un réseau, puisqu'il permet de cibler ensuite des machines. Le scanning d'IP est très souvent suivi par un scanning des ports ouverts sur une machine afin de cibler un service en particulier. L'outil « nmap » est très utilisé pour ce type de scan et compatible IPv6. Pour plus d'information sur cet outil, veuillez se référer au lien suivant : <http://nmap.org/>.

La Figure 72 montre plus en détail comment le script « alive6 » fonctionne :

fe80::8aae:1dff:feaf:4e50	ff02::1	ICMPv6	Echo (ping) request id=0xdead, seq=48879
fe80::8aae:1dff:feaf:4e50	ff02::1	ICMPv6	Echo (ping) request id=0xdead, seq=48879
fe80::f8cd:246b:de87:ac75	ff02::1:ffaf:4e50	ICMPv6	Neighbor solicitation for fe80::8aae:1dff:feaf:4e50 from c8:0a:a9:2a:ec:0f
fe80::8aae:1dff:feaf:4e50	fe80::f8cd:246b:de87:ac75	ICMPv6	Neighbor advertisement fe80::8aae:1dff:feaf:4e50 (sol, ovr) is at 88:ae:1d:af:4e:50
fe80::f8cd:246b:de87:ac75	fe80::8aae:1dff:feaf:4e50	ICMPv6	Parameter problem (Option)

Figure 72 : Paquets ICMPv6 représentant les pings erronés envoyés volontairement par l'attaquant et la réponse de la victime.

Le principe est très simple, pour s'informer des systèmes présents sur le réseau, le script envoie des requêtes « ping » avec un ID erroné (valeur = 0xdead). Les machines recevant ce ping vont donc indiquer un problème de paramètre (Parameter problem) en mentionnant comme adresse IP source leur propre adresse. Le pirate peut ainsi connaître toutes les machines qui répondent, c'est-à-dire toutes celles du réseau local.

## 6.5 Flooding

Cette section présente la façon dont « flooder » une machine victime de fausses adresses IP. Il est ainsi possible pour une personne malveillante de submerger d'adresses IP une victime, ce qui implique les conséquences suivantes pour la ou les victimes :

La première est que la commande « ipconfig » sur la machine victime est totalement inutilisable comme le montre la Figure 73.

```

Adresse IPv6 . . . . . : 2a01:0:9c07:0:f8cd:246b:de87:ac75
Adresse IPv6 . . . . . : 2a01:0:9d00:0:f8cd:246b:de87:ac75
Adresse IPv6 . . . . . : 2a01:0:9d01:0:f8cd:246b:de87:ac75
Adresse IPv6 . . . . . : 2a01:0:9d05:0:f8cd:246b:de87:ac75
Adresse IPv6 . . . . . : 2a01:0:9d06:0:f8cd:246b:de87:ac75
Adresse IPv6 . . . . . : 2a01:0:9d07:0:f8cd:246b:de87:ac75
Adresse IPv6 . . . . . : 2a01:0:9e00:0:f8cd:246b:de87:ac75
Adresse IPv6 . . . . . : 2a01:0:9e01:0:f8cd:246b:de87:ac75
Adresse IPv6 . . . . . : 2a01:0:9e05:0:f8cd:246b:de87:ac75
Adresse IPv6 . . . . . : 2a01:0:9e06:0:f8cd:246b:de87:ac75
Adresse IPv6 . . . . . : 2a01:0:9e07:0:f8cd:246b:de87:ac75
Adresse IPv6 . . . . . : 2a01:0:9f00:0:f8cd:246b:de87:ac75
Adresse IPv6 . . . . . : 2a01:0:9f01:0:f8cd:246b:de87:ac75
Adresse IPv6 . . . . . : 2a01:0:9f05:0:f8cd:246b:de87:ac75

```

Figure 73 : Commande « ipconfig » réalisée après une attaque de type « flooding » et montrant une partie des nombreuses adresses IPv6 enregistrées dans le système.

En plus des multiples adresses IPv6 enregistrées, il y a également des centaines d'adresses IPv6 temporaires ainsi que d'adresses de passerelle par défaut. Une deuxième conséquence directe de cela découle de ce fait puisque, comme expliqué dans la partie « NUD », chaque machine va tester régulièrement la « validité » d'une autre dont il connaît l'adresse. Ainsi, pour chaque adresse de passerelle enregistrée, la victime va générer des « Neighbor solicitation » ce qui va utiliser de la bande passante inutilement.

Une autre conséquence apparaît si un serveur est connecté dans le réseau local dans lequel s'effectue l'attaque. Dans ce cas, le serveur va également apprendre toutes ces adresses IP dues aux « Neighbor advertisement ». Ainsi, si un administrateur veut s'y connecter à distance, il en résulte un temps de connexion extrêmement long qui peut atteindre plusieurs heures. De plus, si ce serveur fait office de serveur DNS, une entrée dans la zone directe va apparaître pour chaque fausse adresse IP comme le montre la capture de la Figure 74.

Name	Type	Data	Timestamp
(same as parent folder)	IPv6 Host (AAAA)	2a01:0b56:eb47:ace9:d939...	21.06.2011 21:00:00
(same as parent folder)	IPv6 Host (AAAA)	2a01:0b5e:b6b4:7276:d939...	21.06.2011 21:00:00
(same as parent folder)	IPv6 Host (AAAA)	2a01:0b8a:e2ac:1772:d939...	21.06.2011 21:00:00
(same as parent folder)	IPv6 Host (AAAA)	2a01:0b8c:e7c0:1cd0:d939...	21.06.2011 21:00:00
(same as parent folder)	IPv6 Host (AAAA)	2a01:0b8f:aabb:8ca6:d939...	21.06.2011 21:00:00
(same as parent folder)	IPv6 Host (AAAA)	2a01:0b91:af50:6442:d939...	21.06.2011 21:00:00
(same as parent folder)	IPv6 Host (AAAA)	2a01:0b98:8ec2:fa67:d939...	21.06.2011 21:00:00
(same as parent folder)	IPv6 Host (AAAA)	2a01:0b9a:e1b4:0ae1:d939...	21.06.2011 21:00:00
(same as parent folder)	IPv6 Host (AAAA)	2a01:0bac:b8dc:5e0c:d939...	21.06.2011 21:00:00
(same as parent folder)	IPv6 Host (AAAA)	2a01:0bb0:c4b9:aecf:d939...	21.06.2011 21:00:00
(same as parent folder)	IPv6 Host (AAAA)	2a01:0bb2:454f:bbe7:d939...	21.06.2011 21:00:00
(same as parent folder)	IPv6 Host (AAAA)	2a01:0bbf:b8b6:1341:d939...	21.06.2011 21:00:00
(same as parent folder)	IPv6 Host (AAAA)	2a01:0bbf:ec93:ddf5:d939...	21.06.2011 21:00:00

Figure 74 : Capture permettant de montrer le « flooding » de la zone directe du DNS.

Ces différents impacts nous montrent à quel point cette attaque peut être destructrice, puisqu'elle atteint non seulement les différents postes Windows 7 mais également les serveurs 2008. Il est évident que le risque de vulnérabilité sur ces derniers reste pour le moins faible, puisqu'ils ne sont jamais placés dans le même réseau que les clients, mais protégés dans des zones démilitarisées.

Cette attaque peut s'effectuer grâce à l'outil « fuzz\_ip6 » du toolkit « thc-ipv6-1.6 » selon la Figure 75.

```
steve@steve-laptop: ~/HEIG-VD/PDB/thc-ipv6-1.6
```

File Edit View Search Terminal Help

```
[steve@steve-laptop thc-ipv6-1.6]$ sudo ./fuzz_ip6 eth0 fe80::f8cd:246b:de87:ac7  
5 -4  
Fuzzing packet, starting at fuzz case 0, ending at fuzz case 1999999999, every p  
acket sent denoted by a dot:  
.....  
.....  
.....  
.....
```

Figure 75 : Capture montrant l'utilisation de l'option 4 du script "fuzz\_ip6" appartenant au toolkit « thc-ipv6-1.6 ».

La commande à lancer est donc « `./fuzz_ip6 <interface> <adresse_victime>` ». Ensuite, des centaines de « Router advertisement » vont être générés par le script afin de « flood » la victime de fausses adresses IPv6 et de fausses passerelles par défaut, pour lesquelles elle devra vérifier la validité. La Figure 76 illustre les nombreux « Neighbor advertisement » envoyés par la machine malveillante.

fe80::2c00:0:feaf:4e50	ff02::1	ICMPv6	Router advertisement from 88:ae:2c:00:00:00
fe80::2d00:0:feaf:4e50	ff02::1	ICMPv6	Router advertisement from 88:ae:2d:00:00:00
fe80::2e00:0:feaf:4e50	ff02::1	ICMPv6	Router advertisement from 88:ae:2e:00:00:00
fe80::2f00:0:feaf:4e50	ff02::1	ICMPv6	Router advertisement from 88:ae:2f:00:00:00
fe80::3000:0:feaf:4e50	ff02::1	ICMPv6	Router advertisement from 88:ae:30:00:00:00
fe80::3100:0:feaf:4e50	ff02::1	ICMPv6	Router advertisement from 88:ae:31:00:00:00
fe80::3200:0:feaf:4e50	ff02::1	ICMPv6	Router advertisement from 88:ae:32:00:00:00
fe80::3300:0:feaf:4e50	ff02::1	ICMPv6	Router advertisement from 88:ae:33:00:00:00

Figure 76 : Paquets ICMPv6 représentant un « flooding » de « Router advertisement » afin de submerger une cible d'adresses IP. On remarque les nombreuses adresses IP et adresses MAC différentes.

**Remarque :** A noter que même si l'adresse victime est précisée (Figure 75), les tests effectués ont montré que toutes les machines présentes sur le réseau local sont impactées. Il est donc fortement déconseillé de reproduire ce type d'attaque sur un réseau comportant un serveur ou un grand nombre de machine, sous peine de passer plusieurs heures à réparer les dégats, les commandes « `ipconfig /release` » et « `ipconfig /renew` » prenant un temps considérable en présence d'un très grand nombre d'IP pour ses interfaces.

```
:: ff02::1:ff ICMPV6 Neighbor solicitation for 2a01:0:a300:0:d9c2:a507:f506:b4c8
:: ff02::1:ff ICMPV6 Neighbor solicitation for 2a01:0:a400:0:f8cd:246b:de87:ac75
:: ff02::1:ff ICMPV6 Neighbor solicitation for 2a01:0:a400:0:d9c2:a507:f506:b4c8
:: ff02::1:ff ICMPV6 Neighbor solicitation for 2a01:0:a500:0:f8cd:246b:de87:ac75
:: ff02::1:ff ICMPV6 Neighbor solicitation for 2a01:0:a500:0:d9c2:a507:f506:b4c8
:: ff02::1:ff ICMPV6 Neighbor solicitation for 2a01:0:a600:0:f8cd:246b:de87:ac75
:: ff02::1:ff ICMPV6 Neighbor solicitation for 2a01:0:a600:0:d9c2:a507:f506:b4c8
:: ff02::1:ff ICMPV6 Neighbor solicitation for 2a01:0:a700:0:f8cd:246b:de87:ac75
```

Figure 77 : Paquets ICMPv6 représentant les « Neighbor solicitation » envoyés par la machine victime aux adresses de passerelles par défaut qu'elle a reçu de l'attaquant. Ceci montre clairement que du trafic réseau est généré inutilement.

## 6.6 Dos-new-IP

Les types d'attaques de déni de service sont multiples et peuvent être appliquées dans beaucoup de cas. Dans la partie « Windows Denial of Service » a été présentée la façon dont « freezer » totalement un poste Windows 7 grâce à des « Router advertisement ». Cette partie présente un autre type de DOS réalisable sur le protocole IPv6. Il s'agit d'empêcher une machine de se connecter au réseau local en l'informant que l'adresse dont elle vérifie l'unicité est déjà utilisée. La machine victime va donc essayer plusieurs tentatives avant d'abandonner et d'afficher le message d'erreur présenté à la Figure 78.

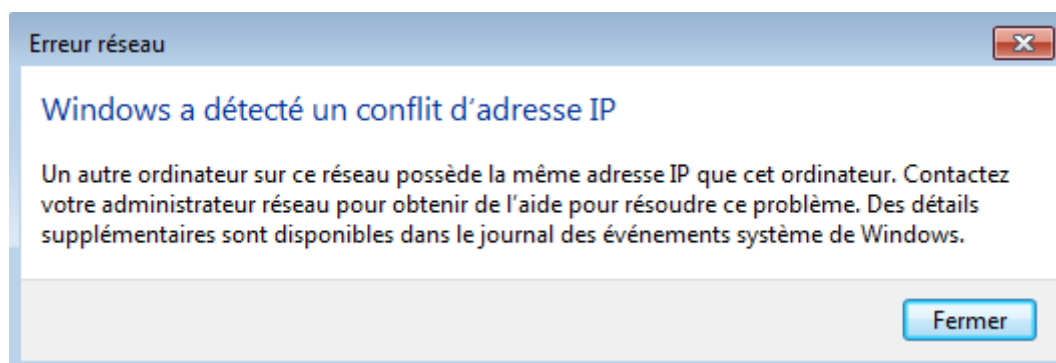


Figure 78 : Message d'erreur affiché lors d'une attaque de type DOS réalisée à l'aide du script « `dos-new-ip6` » du toolkit « `thc-ipv6-1.6` ».

**Remarque :** Pour plus d'explications concernant la détection de validité d'une adresse, se référer à la partie traitant de ce sujet intitulée « DAD ».

Cette attaque est, comme les autres présentées précédemment, très simple à mettre en œuvre avec le toolkit « thc-ipv6-1.6 ». Il suffit de se connecter sur un réseau et de lancer le script « dos-new-ip6 » à l'aide de la commande `sudo ./dos-new-ip6 <interface>`. Une fois la commande lancée et le script exécuté, il est affiché dans la console toutes les adresses qui ont tenté de se connecter au réseau et qui ont échoué, comme le montre la Figure 79.

```
steve@steve-laptop: ~/HEIG-VD/PDB/thc-ipv6-1.6
File Edit View Search Terminal Help
[steve@steve-laptop thc-ipv6-1.6]$ sudo ./dos-new-ip6 eth0
Started ICMP6 DAD Denial-of-Service (Press Control-C to end) ...
Spoofed packet for existing ip6 as fe80::8aae:1dff:feaf:4e50
Spoofed packet for existing ip6 as fe80::f8cd:246b:de87:ac75
Spoofed packet for existing ip6 as 2001:4444:5555::9
Spoofed packet for existing ip6 as fe80::4080:e059:f054:1bb1
Spoofed packet for existing ip6 as fe80::edc4:76ce:5b3b:6256
Spoofed packet for existing ip6 as fe80::c93f:8017:5175:4769
Spoofed packet for existing ip6 as fe80::48a0:ae38:caf9:7407
Spoofed packet for existing ip6 as fe80::8464:1569:f60f:7465
Spoofed packet for existing ip6 as fe80::2d38:eeb4:1696:5f6c
Spoofed packet for existing ip6 as fe80::70cc:cb60:82b:dcdf
Spoofed packet for existing ip6 as fe80::2520:52d9:f3ac:3153
^C[steve@steve-laptop thc-ipv6-1.6]$
```

Figure 79 : Capture montrant l'utilisation du script "dos-new-ip6" appartenant au toolkit « thc-ipv6-1.6 ».

Cette capture montre donc qu'une machine a tenté de se connecter d'abord avec l'adresse `fe80::8aae:1dff:feaf:4e50` puis avec l'adresse `fe80::f8cd:246b:de87:ac75` et ainsi de suite. A chaque fois la machine « attaquante » indique que l'adresse courante est utilisée et de ce fait, la machine cible tente de se connecter avec une nouvelle adresse. Au bout d'une dizaine d'essais, celle-ci abandonne et le message de la Figure 78 est affiché. La victime n'a donc aucun moyen de se connecter au réseau et il en résulte un déni de service.

Si on analyse les actions du script, on remarque qu'il ne fait simplement qu'écouter le trafic réseau et lorsqu'il détecte qu'un paquet « Neighbor solicitation » est envoyé afin de savoir si une adresse est valide, celui-ci répond par la négative grâce à un « Neighbor advertisement » comme le montre la Figure 80.

::	ff02::1:ff	ICMPv6	Neighbor solicitation for fe80::f8cd:246b:de87:ac75
::	ff02::1:ff	ICMPv6	Neighbor solicitation for 2001:4444:5555::9
fe80::ff02::2		ICMPv6	Router solicitation from c8:0a:a9:2a:ec:0f
fe80::ff02::1		ICMPv6	Neighbor advertisement fe80::f8cd:246b:de87:ac75 (ovr) is at 88:ae:fa:0c:2d:64
2001:4ff02::1		ICMPv6	Neighbor advertisement 2001:4444:5555::9 (ovr) is at 88:ae:1a:83:8a:3a
fe80::ff02::1		ICMPv6	Neighbor advertisement fe80::f8cd:246b:de87:ac75 (ovr) is at 88:ae:fa:0c:2d:64
2001:4ff02::1		ICMPv6	Neighbor advertisement 2001:4444:5555::9 (ovr) is at 88:ae:1a:83:8a:3a
fe80::ff02::16		ICMPv6	Multicast Listener Report Message v2
fe80::ff02::16		ICMPv6	Multicast Listener Report Message v2
fe80::ff02::1:ff		ICMPv6	Neighbor solicitation for 2001:4444:5555::1 from c8:0a:a9:2a:ec:0f
fe80::ff02::16		ICMPv6	Multicast Listener Report Message v2
::	ff02::1:ff	ICMPv6	Neighbor solicitation for fe80::4080:e059:f054:1bb1
fe80::ff02::1		ICMPv6	Neighbor advertisement fe80::4080:e059:f054:1bb1 (ovr) is at 88:ae:59:d5:57:47
fe80::ff02::1		ICMPv6	Neighbor advertisement fe80::4080:e059:f054:1bb1 (ovr) is at 88:ae:59:d5:57:47
fe80::ff02::16		ICMPv6	Multicast Listener Report Message v2
fe80::ff02::16		ICMPv6	Multicast Listener Report Message v2
::	ff02::1:ff	ICMPv6	Neighbor solicitation for fe80::edc4:76ce:5b3b:6256
fe80::ff02::1		ICMPv6	Neighbor advertisement fe80::edc4:76ce:5b3b:6256 (ovr) is at 88:ae:36:fc:51:f5
fe80::ff02::1:ff		ICMPv6	Neighbor solicitation for 2001:4444:5555::1 from c8:0a:a9:2a:ec:0f
fe80::ff02::1		ICMPv6	Neighbor advertisement fe80::edc4:76ce:5b3b:6256 (ovr) is at 88:ae:36:fc:51:f5

Figure 80 : Paquets ICMPv6 montrant les « Neighbor solicitation » envoyés par la machine désirant se connecter au réseau et les « Neighbor advertisement » lui indiquant que l'adresse est déjà utilisée.

On remarque donc bien que pour chaque adresse testée (Figure 79) il y a un « Neighbor advertisement » correspondant avec le message « (ovr) is at <adresse MAC> ».

Ce type d'attaque permet de montrer qu'il est possible de réaliser un type de DOS à la connexion de l'utilisateur très simplement.

## 6.7 DHCP snooping

Un problème récurrent en ce qui concerne l'administration de réseau concerne les serveurs DHCP. En effet, il est important du point de vue de la sécurité de pouvoir gérer quel serveur DHCP a le droit de fonctionner sur le réseau. Si tel n'était pas le cas, une personne malveillante pourrait très bien se faire passer pour un faux serveur DHCP. Ceci impliquerait qu'elle pourrait donner aux clients de fausses informations comme par exemple sa propre adresse comme passerelle par défaut. Ainsi, l'attaquant serait placé en « Man in the middle » comme il a été expliqué à la Figure 68. Pour contrer cela, il existe en IPv4 une technique appelée « DHCP snooping » et qui consiste à configurer une liste de ports « trusted » et « non trusted » sur les switchs afin de déterminer quel port a le droit d'émettre des paquets « DHCP OFFER » et « DHCP ACK ». Par exemple, les commandes suivantes autorisent l'interface « fastethernet 0/0 » à émettre ce type de paquets :

1. *Switch(config)# ip dhcp snooping*
2. *Switch(config)# ip dhcp snooping vlan 1*
3. *Switch(config)# interface fastethernet0/0*
4. *Switch(config-if)# ip dhcp snooping trust*



La première active la fonctionnalité « snooping » sur le switch, la seconde définit pour quel VLAN le snooping sera activé et enfin, les troisième et quatrième permettent de définir quel port sera un port de confiance dans le cadre du service DHCP.

En IPv6, la fonction « snooping » n'est pas activable tel quel grâce aux commandes ci-dessus. Pour se protéger de ce type d'attaque, il y a la possibilité, sur les switches « layer 3 » d'accepter uniquement les paquets « DHCPv6 » provenant du serveur, en fonction de son adresse IP. Si on se base sur l'architecture de la Figure 81, les ports sur lesquels seront connectés des « End Node » doivent obligatoirement bloquer tout paquet de type « DHCPv6 Advertise » pour ainsi éviter à un client malveillant de se faire passer pour un serveur DHCP envers les autres clients connectés.

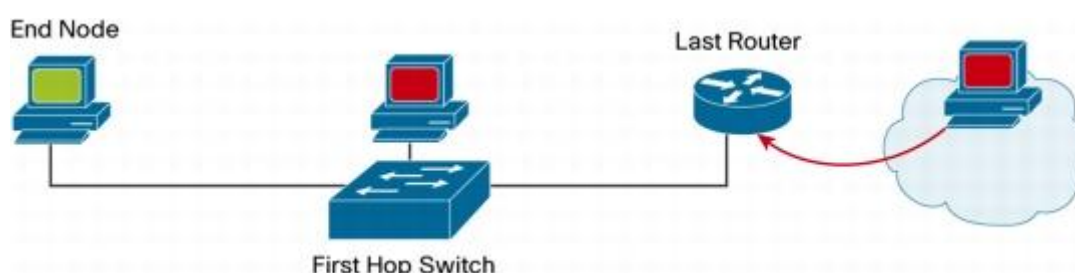


Figure 81 : Illustration d'une architecture typique dans laquelle les ports des « End Nodes » ne doivent pas laisser passer des paquets « DHCPv6 Advertise ».

(Source : <http://www.cisco.com>)

Pour implémenter ce type de protection, Cisco préconise de définir une « access list » selon les commandes suivantes :

1. `ipv6 access-list ACCESS_PORT`
2. `deny udp any eq 547 any eq 546`
3. `permit any any`

Ces 3 commandes permettent de refuser tous les paquets UDP ayant un port source valant 547 et un port de destination valant 546. Ceci implique que les clients connectés aux ports sur lesquels seraient appliqués cette « access list » ne pourraient pas transmettre leur « DHCP Avertise » mais pourraient par contre en recevoir. Pour appliquer cette règle à un port il suffit d'entrer les commandes suivantes, relatives à l'interface « gigabitEthernet 0/0 » :

1. `interface gigabitethernet 0/0`
2. `switchport`
3. `ipv6 traffic-filter ACCESS_PORT in`

Tout comme en IPv4, les « access list » permettent de très bien configurer les droits des différents éléments faisant partie du réseau. Une autre utilisation que l'on peut tirer de celles-ci et qui permettrait de parer certaines des attaques présentées précédemment consisterait à en définir sur les ports où sont connectés les « End Node » afin de bloquer les « Router advertisement ». Les commandes suivantes permettent de le faire de la même manière que les précédentes relatives au DHCP :

1. `ipv6 access-list ACCESS_PORT`
2. `deny icmp any any router-advertisement`
3. `permit any any`

Ainsi, tous les ports sur lesquels on définit cette « access list » bloqueront les « Router advertisement » dans un sens, ce qui permet d'empêcher une personne malveillante d'inonder le réseau et de réaliser l'une des attaques précédentes. A noter que sur certains types de switch, notamment les modèles Catalyst 6500 and 4500 Series, Cisco a prévu un raccourci des commandes suivantes pour bloquer ces paquets :

1. `interface gigabitethernet 1/0/1`
2. `switchport`
3. `ipv6 nd raguard`

Malgré les parades présentées, le DHCP snooping reste non-implémenté pour IPv6 comme il l'était pour IPv4. C'est donc à l'administrateur du réseau de s'assurer de définir les bonnes « access-list » sur tous les ports qui ne doivent pas relayer de paquet DHCP. En plus du fait que s'il ne le fait pas, une personne malintentionnée pourrait s'adonner à des attaques contre les machines du réseau, il n'est pas rare de voir des ordinateurs ou des modems ayant leur propre serveur DHCP actif. Par exemple, si le partage de connexion est autorisé sur un ordinateur doté de Windows 7, un DHCP est forcément activé dessus. Les utilisateurs ne sont donc pas forcément au courant qu'ils mettent en péril la sécurité et le bon fonctionnement du réseau. Les problèmes liés aux DHCP « fantômes » sont donc à prendre très au sérieux, que l'on utilise l'ancienne ou la nouvelle norme.

## 6.8 Secure Neighbor Discovery

Les différentes attaques présentées dans ce chapitre relevant différentes failles de sécurité propres à IPv6 (mais dont certaines sont semblables en IPv4) s'appuient toutes sur les faiblesses du protocole NDP. Pour se prémunir contre ceci, le protocole « SEND » pour « Secure Neighbor Discovery » a été mis en place. Cette sous-section présente un aperçu des différentes spécificités de ce protocole très important en termes de sécurité. A noter qu'une autre façon de bloquer certaines de ces attaques, basée sur des « access list », a été présentée à la fin de



la partie « DHCP snooping ».

**Remarque :** Cette partie est purement théorique, et n'a pas fait l'objet de tests dans le cadre de ce travail.

### 6.8.1 Adresses CGA

Tout d'abord, ce protocole utilise des adresses dites « CGA » (Cryptographically Generated Addresses) qui sont définies selon la RFC 3971. Celles-ci sont un type d'adresse IPv6 unicast qui permet de lier l'identifiant d'interface de chaque adresse avec une clé publique. Sans entrer dans les détails, ceci permet en particulier de prouver l'authenticité des routeurs d'accès et protège contre l'usurpation d'adresse.

### 6.8.2 Options supplémentaires

Le protocole SEND définit six options supplémentaires permettant de sécuriser le protocole NDP.

- **CGA :** Contient les paramètres propres aux adresses CGA.
- **Horodatage et Nonce :** Permet d'éviter les attaques par « rejeu ».
- **Signature RSA :** Contient la signature RSA du message.
- **Certificat :** Contient un certificat à envoyer à l'hôte dans le cas où un message est de type CGA.

Ces 4 options ajoutées au protocole NDP apportent les avantages suivants en termes de sécurité :

- Il est impossible de faire de l'« IP spoofing » sur le protocole, l'identifiant d'interface étant lié à l'adresse IP grâce aux adresses CGA. L'attaque « Man in the middle » présentée précédemment n'est donc pas applicable si le protocole SEND est utilisé.
- L'intégrité des messages envoyés est garantie par leur signature RSA. Il n'est donc pas possible de modifier des paquets transitant sur le réseau sans que la destination s'en aperçoive.
- Les attaques par « rejeu » sont inefficaces, grâce au champ « horodatage » et « nonce ». On ne peut donc pas capturer un message d'authentification dans le but de le « rejouer » ensuite pour pouvoir s'authentifier avec la session d'un autre.

Etant donné la complexité du protocole, il est inutile d'entrer plus dans les détails dans le cadre de ce travail, celui-ci amenant une vision basée sur la pratique plutôt que la théorie. Il est cependant important de prendre conscience que le NDP n'est pas du tout sécurisé et que les sites sensibles doivent communiquer à l'aide du SEND pour garantir les différents points ci-dessus et éviter tout type d'attaques. Malheureusement, aucune version de Windows ne supporte ce protocole à l'heure actuelle. Par contre, les équipements Cisco récents sont pour la plupart compatibles et disposent de commandes prévues à cet effet.

**Remarque :** Pour toutes informations complémentaires concernant ce protocole, veuillez se référer à la thèse de M.Tony Cheneau à l'adresse suivante : <http://amnesiak.org/me/papers/thesis-2011.pdf>.

Les informations concernant la configuration du protocole SEND sont disponibles à l'adresse : [http://www.cisco.com/en/US/docs/ios/ipv6/configuration/guide/ip6-first\\_hop\\_security.html](http://www.cisco.com/en/US/docs/ios/ipv6/configuration/guide/ip6-first_hop_security.html).

## 6.9 Cisco ASA

Lors de l'étude de la sécurité sur IPv6, le produit de la gamme « ASA » proposée par Cisco, qui permet de mettre en place des firewalls et des passerelles VPN, a également été testée afin de donner un meilleur aperçu des fonctionnalités qu'il est possible d'implémenter avec cette norme. L'idée de cette partie est donc de montrer une vue d'ensemble des compatibilités qu'a l'ASA avec IPv6.

Premièrement, le gros avantage qu'offre l'ASA, indépendamment de la norme IP utilisée, est sa configuration rendue plus simple par une interface graphique claire. C'est principalement cela qui fait son succès dans les entreprises. Il est ainsi beaucoup plus aisé de configurer son firewall ou encore son accès VPN grâce à des règles « graphiques » plutôt qu'en utilisant une « CLI » (Command line interface), qui peut s'avérer compliquée dans le cadre de ce type de configuration. Les commandes exécutées sur l'interface graphique, sont ensuite traduites en commandes Cisco pour simplifier la tâche de l'administrateur.

Pour ce qui est de sa compatibilité avec IPv6, la configuration de base de l'ASA l'est entièrement. Il est donc possible de le brancher sur un réseau IPv6, en définissant des adresses IPv6 sur les interfaces. Il permet également de faire office de serveur DHCPv6. En tant que firewall, il est possible de définir des règles dans les deux normes. Les principaux protocoles comme ICMPv6 et DHCPv6 sont implémentés et des règles peuvent être appliquées en se basant sur ceux-ci. Malgré cela, aucune des architectures VPN, que ce soit du « Site-to-Site » ou du « Remote Access » ne sont compatibles avec IPv6. Il est pour l'instant impossible de créer un tunnel VPN en IPv6 en utilisant ce type de plateforme. Pour se faire, il est préférable de se tourner vers le produit « Direct Access » de Microsoft, présenté dans la partie suivante.

L'ASA offre également un autre avantage, qui pourra être utilisé dans le cadre d'IPv6. Il y a en effet la possibilité de le faire communiquer avec un agent à installer sur l'AD ce qui a le mérite d'offrir des fonctionnalités supplémentaires. L'administrateur peut donc définir des règles directement en fonction des utilisateurs plutôt que par rapport à leurs adresses IP. Avec la nouvelle norme, cela prend tout son sens, puisque les utilisateurs pourront en avoir plusieurs (locales, globales, temporaires, etc...). En plus de cela, la mobilité des utilisateurs est bien plus facile à gérer puisque même si l'utilisateur change d'adresse IP, la règle définie sur le « Firewall » sera

toujours valide. Du point de vue d'un administrateur, il est également plus simple de définir des règles par rapport aux utilisateurs plutôt qu'à leurs adresses. A noter que la définition de règles par rapport à des groupes d'utilisateurs est également possible.

Pour illustrer le fonctionnement de base de cette architecture, le schéma de la Figure 82.

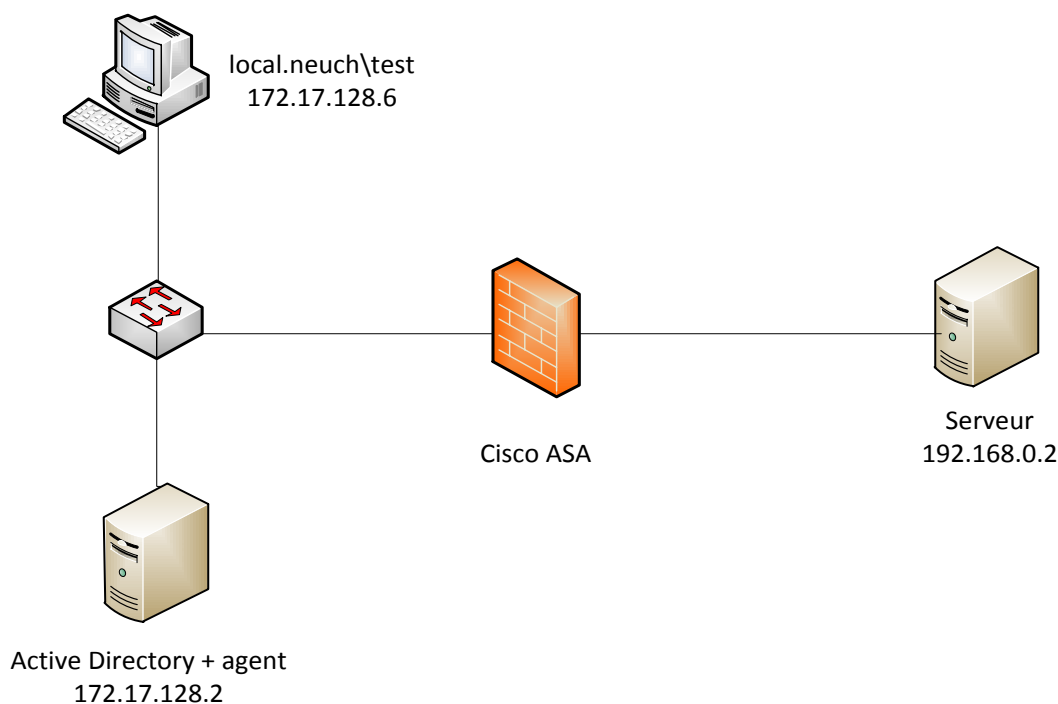


Figure 82 : Schéma illustrant le principe de base d'une architecture où l'ASA communique avec l'Active Directory par l'intermédiaire d'un agent installé sur celui-ci.

Cette architecture très simple permet à l'utilisateur « test » du domaine « local.neuch », dont l'Active Directory fait également office de contrôleur de domaine, de « ping » le serveur situé derrière le firewall. Les deux règles normalement définies sur le firewall permettraient à l'adresse IP 172.17.128.6 d'utiliser le protocole « ICMP » vers l'adresse 192.168.0.2. Cela se passe différemment si l'ASA est correctement configuré pour communiquer avec l'AD. A la connexion de l'utilisateur sur le domaine « local.neuch », l'agent va établir un « mapping » entre son adresse IP et son nom d'utilisateur que l'ASA va pouvoir consulter lorsqu'il est confronté à une règle définie sur un nom d'utilisateur. Ce fonctionnement apporte donc tous les avantages présentés précédemment. A noter qu'il y a également la possibilité de logger les connexions des utilisateurs sur un serveur « syslog » à des fins d'analyse. Les tests ont été effectués dans les deux normes, l'agent étant compatibles dans tous les cas.

***Remarque :** Une configuration des différents éléments de cette architecture est présentée en annexe afin de pouvoir la reproduire facilement. Il est à noter également que c'est à l'administrateur réseau de l'adapter en fonction de ses besoins, l'AD n'étant jamais situé dans le même sous-réseau que les clients pour des raisons évidentes de sécurité.*

## 6.10 Microsoft DirectAccess

De nos jours, les accès à distance à l'intranet d'une entreprise deviennent indispensables pour le bon fonctionnement de celle-ci. En effet, entre les employés devant accéder depuis n'importe où à un serveur de fichier, ceux devant centraliser une ressource sur l'intranet, ou encore les administrateurs réseau devant configurer un élément du réseau, il est primordial de pouvoir offrir à toutes ces personnes le moyen d'établir une connexion sécurisée entre le lieu où ceux-ci vont l'initier et le site de l'entreprise. Les solutions utilisées dans ces cas-là sont les tunnels VPN (Virtual Private Network). Pour rappel, ces tunnels peuvent être de plusieurs types, qu'on peut classer selon les 2 catégories suivantes :

- Site-to-Site

Le tunnel sécurisé est établi d'un site à un autre, c'est-à-dire que les clients d'un site d'une entreprise accèdent de manière sécurisée à un autre site comme si les deux sites étaient au même endroit. Les clients utilisant le VPN peuvent accéder de manière sécurisée à l'intranet, **pour autant qu'ils se trouvent dans un site prévu à cet effet**. Ils ne pourraient donc pas établir de connexion sécurisée depuis un autre endroit. Cette façon de faire a pour avantage de pouvoir diviser géographiquement les ressources d'une entreprise sans pour autant priver les utilisateurs de celles qui ne se trouveraient pas directement sur le site dans lequel ils sont.

- Remote Access

A l'inverse du mode « Site-to-Site », cette configuration établit un tunnel sécurisé **entre une machine et un site distant**. Ainsi, la personne possédant des accès VPN de type « remote access » peut, partout dans le monde, accéder à l'intranet de son entreprise. Le principe de base est que la machine voulant établir une connexion sécurisée avec le site distant contacte une « VPN gateway » placée en frontal du réseau local. Celle-ci peut ensuite déterminer si la machine a le droit d'accéder au réseau. Les avantages de cette manière de faire comparés au type « Site-to-Site » précédent est qu'elle permet une mobilité totale des clients. Par contre, une configuration spécifique doit être effectuée sur toutes les machines clientes qui doivent pouvoir accéder au VPN.

Il existe à ce jour plusieurs logiciels faisant office de client VPN, libres ou propriétaires, et également des plateformes pouvant faire office de « VPN gateway ». Le produit qui va être présenté dans ce chapitre est « Direct Access » de Microsoft.

Celui-ci permet selon le schéma de la Figure 83, d'établir une connexion sécurisée de type « Remote Access » entre un client et un réseau local.

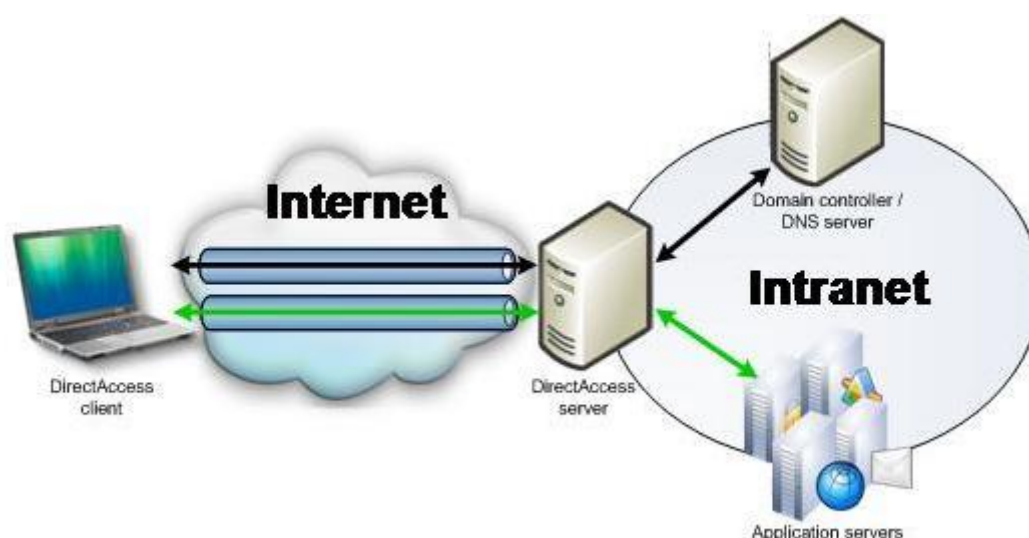


Figure 83 : Schéma présentant l'utilisation du service Direct Access proposé par Microsoft pour établir des connexions sécurisées à distance vers l'intranet d'une entreprise.

(Source : <http://technet.microsoft.com>)

La particularité de « Direct Access » est qu'il est uniquement compatible IPv6 et utilise également le protocole IPsec, situé juste au-dessus d'IP dans la pile de protocole OSI. L'utilisation d'IPsec comprend donc l'authentification de l'ordinateur et de l'utilisateur, ainsi que le chiffrement des données (encryption). Ce qu'apporte ce produit en plus des autres solutions VPN disponibles (pour autant que celles-ci soient compatibles IPv6), est la mise en relation du serveur Direct Access avec l'Active Directory. Ainsi, l'administrateur du serveur a la possibilité de définir quel objet (utilisateur ou groupe) de l'AD a le droit de communiquer avec quelle ressource. Il a en outre les choix suivants :

- **Accès total**

Donne la possibilité au client d'accéder à toutes les ressources de l'intranet. Les données sont chiffrées uniquement lorsqu'elles traversent Internet. Une fois qu'elles arrivent dans le réseau local, elles ne le sont plus (équivalent au mode « tunnel » d'IPsec).

- **Accès sélectif**

Donne la possibilité au client d'accéder aux ressources auxquelles il a le droit d'accéder. Ceci est bien sûr défini par l'administrateur du réseau.

- **Accès sécurisé**

Donne la possibilité au client d'accéder de bout en bout à un serveur. A la différence du premier cas de figure (accès total), les données sont cette fois-ci chiffrées également dans l'intranet, jusqu'à la machine destinataire. Ce type d'accès équivaut au mode « transport » d'IPsec.

Un autre avantage apporté par ce produit est la gestion simplifiée du parc d'ordinateurs, puisque les mises à jour des GPO peuvent être transmises au client dès que celui-ci se connecte au VPN. Ceci garanti donc que tous les clients de l'entreprise soient à jour en permanence, ce qui constitue un avantage certain du point de vue de la sécurité d'une entreprise.

Sur le client, Direct Access n'impose pas de logiciel client, et la connexion se fait par le « gestionnaire de connexion » de Windows 7. Il est donc aisé de paramétrer un grand nombre de clients très simplement, du fait qu'aucun logiciel supplémentaire ne soit requis.

De manière simplifiée, la connexion s'effectue comme suit : Une fois que le client est connecté à un réseau, il va tenter de contacter le serveur Direct Access et de s'y connecter en IPv6 avec IPsec. Si la connexion est établie, les deux s'authentifient grâce à des certificats et le serveur contrôle ensuite que le client appartient à un groupe autorisé à se connecter à l'intranet. Une infrastructure PKI est donc indispensable pour le bon fonctionnement du serveur Direct Access. L'installation de celle-ci est détaillée en annexe, dans la partie *Installation du service « Direct Access »*.

## 7 Conclusion

Au terme de ce travail, il est nécessaire de faire un bilan des points principaux qui peuvent être retirés, afin de donner au lecteur une vision globale du travail.

Tout d'abord, le changement majeur de la norme vient de l'adressage, codé sur 128 bits, ce qui représente un espace d'adressage énorme. Celui-ci définit également des nouveaux types d'adresses, ayant chacun leurs avantages : les adresses de site, « link locales », anycast. Les adresses de site ont la particularité de pouvoir faire communiquer une machine uniquement sur un site, celles-ci étant non routables sur Internet. Des serveurs internes à l'entreprise et qui ne doivent pas être accédés par l'extérieures peuvent se voir assigner des adresses de ce type. Le choix est laissé aux administrateurs du réseau de les utiliser ou pas, tout en tenant compte que celles-ci devraient disparaître de la prochaine RFC\*. Les adresses « link locales » apportent l'avantage de pouvoir faire communiquer des machines sans aucune configuration préalable au sein d'un réseau local. Ces adresses ne sont donc pas routables. Enfin, les adresses anycast permettent à une machine de contacter le service « le plus proche » ce qui améliore grandement les performances du réseau.

Le protocole ICMPv6 apporte les nouvelles fonctionnalités qui sont les suivantes :

- **Traduction d'adresse**

Celle-ci est similaire au mécanisme d'ARP connu en IPv4. Par contre, si la machine ne connaît pas son préfixe réseau (dans le cas d'une configuration « DHCP statefull »), elle est obligée de passer par l'adresse MAC du routeur pour envoyer ses paquets.

- **DAD (Duplicated Address Detection)**

Cette fonction permet à une machine de détecter si l'adresse qu'elle veut s'auto-configurer n'est pas déjà utilisée.

- **NUD (Neighbor Unreachability Detection)**

Enfin, le NUD permet à une machine de connaître en permanence l'état de ses voisins, et au cas où ils ne sont plus accessibles, de les effacer de son cache.

Ces fonctionnalités font toute partie du protocole NDP.

*\*Selon plusieurs sources dont Mme Silvia Haggren, spécialiste IPv6.*

Le mécanisme de découverte de MTU est également assuré par le protocole ICMPv6 et est entièrement similaire à ce qui existe en IPv4, à part le fait que les routeurs n'ont plus la possibilité de fragmenter les paquets si ceux-ci sont plus grands que le MTU de l'interface concernée.

Du point de vue du service DHCP, celui-ci devient DHCPv6 dans la nouvelle norme et définit trois modes différents : Le statefull, le stateless, et l'auto-configuration. Le mode statefull ne présente aucun changement par rapport à l'ancienne norme. Il est utilisé lorsqu'une machine contacte un DHCP pour récupérer son adresse et ses options de configuration. Le second mode est défini lorsque la machine s'auto-configure une adresse par rapport au préfixe réseau que le routeur lui envoie, et va chercher ses options sur le serveur DHCP. Enfin le dernier mode permet à une machine d'auto-configurer son adresse grâce au préfixe réseau envoyé par le routeur, sans avoir une quelconque communication avec le DHCP.

Le service DNS ne présente aucun changement par rapport à celui défini en IPv4, si ce n'est le fait que les enregistrements sont nommés « AAAA » par rapport à « A » dans l'ancienne norme. Les requêtes DNS effectuées par les ordinateurs Windows 7 se font d'abord en IPv4 et ensuite en IPv6.

Comme tous les nouveaux produits sortant sur le marché, la sécurité d'IPv6 comporte des failles qu'il est nécessaire de connaître et de comprendre. Il est ainsi possible de réaliser le même genre d'attaque en local qu'en IPv4, comme le « Man in the middle », des denis de service et du scanning. Il existe cependant des solutions permettant de sécuriser un maximum les réseaux locaux contre ce type d'attaque. L'une d'entre elle provient du protocole SEND qui permet entre autre d'éviter les trois types d'attaques cités en utilisant des concepts comme les certificats, et des signatures créées à partir des adresses MAC des interfaces. Les attaques de types DHCP snooping peuvent également être bloquées grâce à des « access list » correctement configurées sur les ports des routeurs afin de bloquer les paquets DHCP provenant des ports « non trusted ».

Certaines entreprises comme Microsoft ou Cisco ont également prévu des produits compatibles avec la nouvelle norme, bien que tous leurs équipements ne le soient pas. Le Direct Access de Microsoft permet un accès sécurisé en IPv6 au réseau local de l'entreprise par un VPN. L'avantage qu'il propose par rapport aux autres solutions est qu'il est entièrement lié à l'Active Directory. Il est ainsi possible de définir des droits d'accès en fonction des objets de l'AD, comme les utilisateurs ou les groupes.

Le produit ASA de Cisco est aussi compatible avec IPv6, dans sa configuration de base et celle de son firewall. Par contre, les accès VPN qu'ils proposent ne sont pas prévus pour la nouvelle norme. Il est donc impossible pour l'instant de configurer la gateway VPN IPv6. Par contre, il est possible de le lier avec l'Active Directory par l'intermédiaire d'un plugin qui permet de définir des règles sur le firewall propres aux utilisateurs, et non plus aux adresses IP. Ceci a l'avantage de simplifier les règles à définir, notamment en pouvant définir des règles sur des



groupes, mais également de laisser une plus grande mobilité à l'utilisateur, sans pour autant prévoir de règle supplémentaire.

Concernant l'implémentation d'IPsec dans IPv6, le choix d'utiliser ce protocole est laissé à l'application en elle-même. Les échanges IPv6 de base ne sont pas chiffrés par défaut, mais les champs pour le faire sont déjà prévus par la norme.

Ce travail a donc permis d'apporter les éléments nécessaires au SIEN afin qu'il puisse se pencher sérieusement sur la norme à l'avenir tout en ayant déjà une vision de ce qu'il est possible de faire et de ce qu'il ne l'est pas. Comme mentionné dans l'introduction de ce travail, cette migration ne doit pas se faire du jour au lendemain, mais de manière réfléchie pour chaque nouveau projet, en intégrant directement la « problématique » IPv6 à tout développement, que ce soit de produits, de services ou de nouvelles infrastructures. Les problèmes de compatibilité viendront les uns après les autres, et pourront être résolus lors de tests préalables à l'installation définitive de la norme.

Personnellement, ce travail m'a beaucoup motivé puisque j'ai été très intéressé de travailler sur une norme émergente comme IPv6. Le côté pratique du travail m'a beaucoup plu et m'a permis de pouvoir toucher à de nombreux domaines comme les serveurs Windows ainsi que les routeurs Cisco, ou encore le Direct Access et l'ASA. Je pense que ces connaissances me seront très utiles pour mon avenir, d'une part parce que ces produits sont très utilisés dans les entreprises, et d'autre part car la norme IPv6 n'est pas encore très connue de tous et que cela peut constituer un avantage certain dans le domaine.

Par rapport au cahier des charges, je pense avoir atteint les objectifs demandés en essayant de développer le plus possible les différents points afin que cela puisse aider au maximum le SIEN dans ses travaux futurs. J'espère donc que mon travail leur sera utile et qu'il constituera un bon point de départ de la migration vers IPv6.

## 8 Glossaire

**AD** : Active Directory

**ARP** : Address Resolution Protocol

**ASA** : Adaptive Security Appliance

**CA** : Certification Authority

**CGA** : Cryptographically Generated Addresses

**CLI** : Command Line Interface

**DAD** : Duplicate Address Detection

**DHCP** : Dynamic Host Configuration Protocol

**DNS** : Domain name system

**DOS** : Denial Of Service

**GPO** : Group Policy Object

**ICMP** : Internet Control Message Protocol

**IP** : Internet Protocol

**MAC** : Media Access Control

**MTU** : Maximum Transmission Unit

**NDP** : Neighbor Discovery Protocol

**NUD** : Neighbor Unreacheable Discovery

**OSI** : Open Systems Interconnection

**PKI** : Public Key Infrastructure

**RAM** : Random Access Memory

**RSA** : Rivest Shamir Adleman (Algorithme de cryptographie asymétrique)

**SOA** : Start Of Authority

**SEND** : Secure Neighbor Discovery

**SIEN** : Service Informatique de l'état de Neuchâtel

**TR** : Télécommunication, Réseaux et services

**UDP** : User Datagram Protocol

**VLAN** : Virtual Local Area Network

**VPN** : Virtual Private Network

## i. Sources

### i. Bibliographie

- <http://www.misfu.com/telechargement-cours-186.html>
- [http://idum.fr/IMG/pdf/Cours\\_IPv6pdf.pdf](http://idum.fr/IMG/pdf/Cours_IPv6pdf.pdf)
- Tomorrow IPv6 – Are you ready ? de M. Fabien Bruchez, LANexpert
- Cours de PDR de M. Stephan Robert, HEIG-VD
- IPv6 Network Administration de Niall Richard Murphy & David Malone
- IPv6 Essentials de Mme Silvia Hagen

### ii. Webographie

#### i. Documentation IPv6

- <http://www.6diss.org/>
- <http://www.labo-microsoft.org/articles/passage-ipv4-ipv6-windows-vista/0/>
- [http://livre.g6.asso.fr/index.php/Main\\_Page](http://livre.g6.asso.fr/index.php/Main_Page)

#### ii. Windows et Ipv6

- <http://technet.microsoft.com/en-us/network/bb530961.aspx>
- <http://technet.microsoft.com/en-us/network/cc987595.aspx>
- <http://research.microsoft.com/en-us/projects/msripv6/>
- <http://technet.microsoft.com/en-us/magazine/cc135901.aspx>
- <http://blogs.technet.com/b/ipv6/>

#### iii. Configuration DHCP sous Windows

- [http://www.labo-microsoft.org/articles/IPv6\\_WindowsServer/3/Default.asp](http://www.labo-microsoft.org/articles/IPv6_WindowsServer/3/Default.asp)
- <http://new-tech.over-blog.com/article-installation-d-un-serveur-dhcp-win2008-assistant-powershell-48434181.html>
- [http://www.toutwindows.com/ws2008r2\\_dhcp.shtml](http://www.toutwindows.com/ws2008r2_dhcp.shtml)

## iv. Configuration DNS sous Windows

- [http://www.labo-microsoft.org/articles/win/delegation\\_dns/2/](http://www.labo-microsoft.org/articles/win/delegation_dns/2/)

## v. Enregistrement dynamique DNS

- <http://technet.microsoft.com/fr-fr/library/cc753014%28WS.10%29.aspx>
- <http://support.microsoft.com/kb/945397/fr>
- <http://technet.microsoft.com/fr-fr/library/cc771255.aspx>
- <http://technet.microsoft.com/fr-fr/library/cc770822.aspx>

## vi. Configuration DNS avec DHCP

- <http://technet.microsoft.com/fr-fr/library/cc787034%28WS.10%29.aspx>

## vii. Commande netsh

- <http://www.winmgr.com/?p=415>

## viii. Installation du service « Active Directory »

- <http://mtodorovic.developpez.com/tutoriels/windows/installation-active-directory-sous-windows-server-2008-r2/>

## ix. Installation du service « Direct Access »

- [http://www.labo-microsoft.org/articles/Direct\\_Access/1/](http://www.labo-microsoft.org/articles/Direct_Access/1/)
- <http://technet.microsoft.com/en-us/library/ee649180%28WS.10%29.aspx>

## x. Sécurité IPv6

- [http://www.v6summit.com/Tutorial/CLASSROOMTUTORIAL\\_ROUTAGE\\_SECURITY.pdf](http://www.v6summit.com/Tutorial/CLASSROOMTUTORIAL_ROUTAGE_SECURITY.pdf)
- <http://pacsec.jp/psj05/psj05-vanhauser-en.pdf>
- <http://www.6net.org/events/workshop-2003/marin.pdf>

- <http://blogs.cisco.com/security/ipv6-security-testing/>
- <http://ipv6.com/articles/research/Secure-Neighbor-Discovery.htm>
- [http://www.cisco.com/en/US/prod/collateral/iosswrel/ps6537/ps6553/whitepaper\\_c11-602135.html](http://www.cisco.com/en/US/prod/collateral/iosswrel/ps6537/ps6553/whitepaper_c11-602135.html)

## **xi. Cisco ASA**

- [http://www.cisco.com/en/US/docs/security/ibf/release\\_notes/ibf10\\_rn.html](http://www.cisco.com/en/US/docs/security/ibf/release_notes/ibf10_rn.html)
- [http://www.ciscosystems.com/en/US/docs/security/ibf/setup\\_guide/ibf10\\_setup\\_guide.pdf](http://www.ciscosystems.com/en/US/docs/security/ibf/setup_guide/ibf10_setup_guide.pdf)

## **xii. Documentation et Configuration de THC-toolkit**

- <http://www.thc.org/thc-ipv6/>
- <http://www.tux-planet.fr/la-faille-dos-ipv6-neighbor-discovery-pour-windows/>

## **xiii. Configuration DHCP Cisco**

- <http://www.cisco.com/en/US/docs/ios/ipv6/configuration/guide/ip6-dhcp.html#wp1054245>
- [http://www.tech-recipes.com/rx/174/configure\\_cisco\\_router\\_as\\_basic\\_dhcp\\_server/](http://www.tech-recipes.com/rx/174/configure_cisco_router_as_basic_dhcp_server/)

## **xiv. Secure Neighbor Discovery Protocol**

- [http://www-public.it-sudparis.eu/~lauren\\_m/articles/Cheneau-SARSSI2010.pdf](http://www-public.it-sudparis.eu/~lauren_m/articles/Cheneau-SARSSI2010.pdf)
- <https://www-public.it-sudparis.eu/~cheneau/papers/article-SAR-SSI-2009.pdf>
- <http://amnesiak.org/me/papers/thesis-2011.pdf>

## A. Annexes

### A. Configuration DHCPv6

Cette partie explique au lecteur désirant reproduire les différentes manipulations présentées dans ce rapport comment configurer un serveur DHCPv6 sous Windows server 2008. Des captures d'écran sont là pour l'aider à la compréhension de la configuration.

Une fois le service DHCP sélectionné dans les rôles du serveur, nous arrivons sur la fenêtre suivante où il est demandé le nom de la nouvelle zone et sa description :

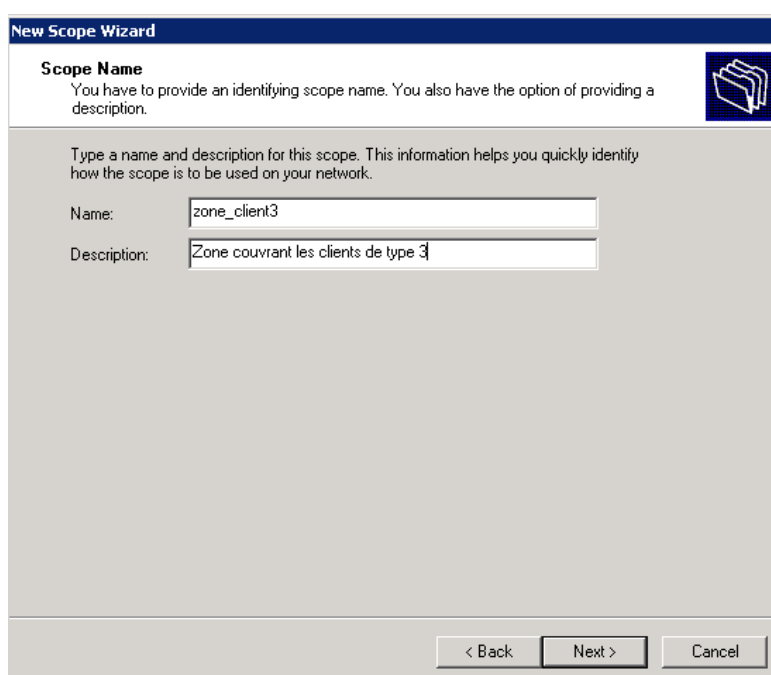
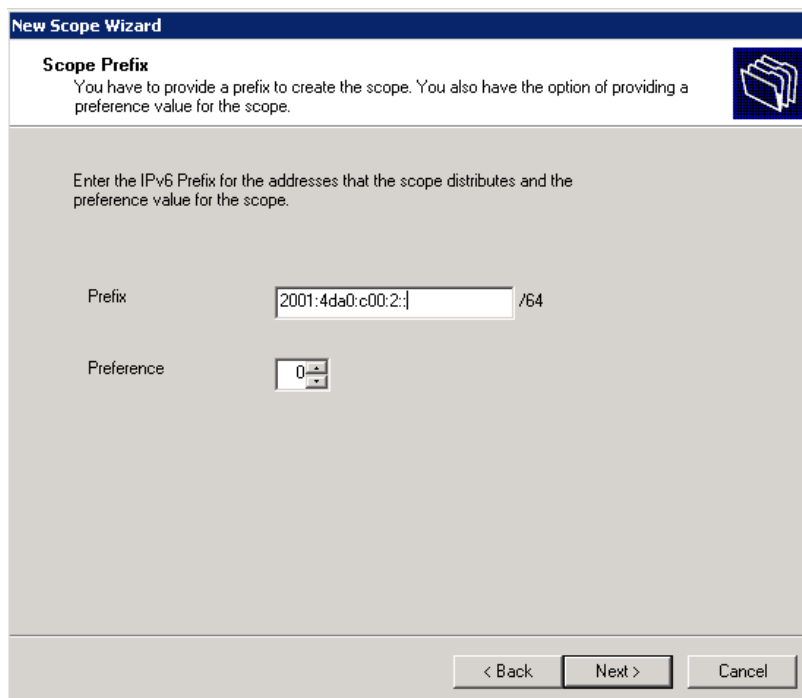


Figure 84 : Configuration DHCP - Nom

Ensuite, il est demandé d'indiquer la valeur du préfixe réseau de la zone :



**New Scope Wizard**

**Scope Prefix**  
You have to provide a prefix to create the scope. You also have the option of providing a preference value for the scope.

Enter the IPv6 Prefix for the addresses that the scope distributes and the preference value for the scope.

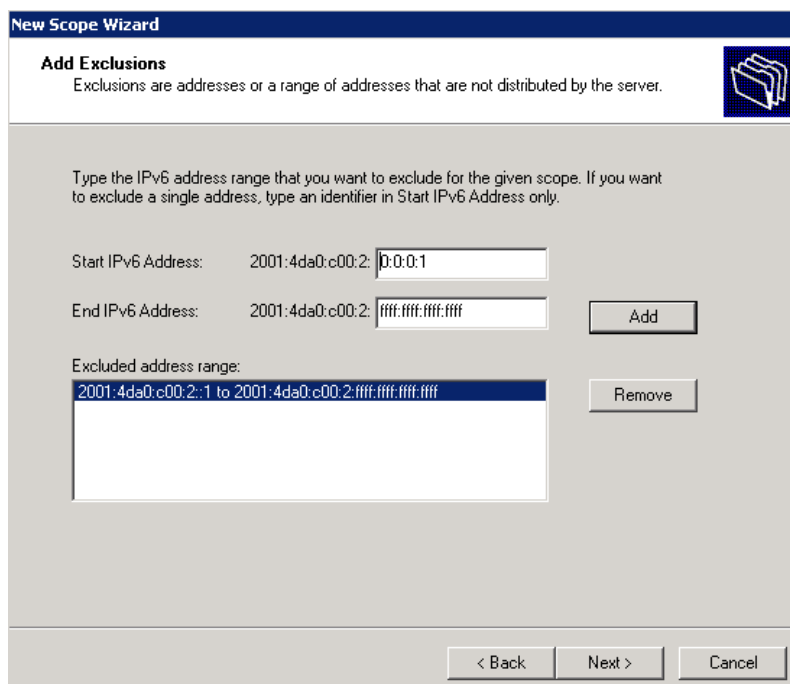
Prefix:  /64

Preference:

< Back Next > Cancel

Figure 85 : Configuration DHCP - Préfixe

Si des adresses doivent être exclues de l'ensemble des adresses pouvant être attribuées à un client, ceci se fait de la manière suivante :



**New Scope Wizard**

**Add Exclusions**  
Exclusions are addresses or a range of addresses that are not distributed by the server.

Type the IPv6 address range that you want to exclude for the given scope. If you want to exclude a single address, type an identifier in Start IPv6 Address only.

Start IPv6 Address:

End IPv6 Address:

Excluded address range:

< Back Next > Cancel

Figure 86 : Configuration DHCP - Exclusion



Finalement, nous pouvons encore choisir la durée de validité d'une adresse.

Figure 87 : Configuration DHCP – Durée de validité

La figure suivante nous montre le résumé des paramètres choisis lors des étapes précédentes, que nous pouvons valider en cliquant sur « Finish ».

Figure 88 : Configuration DHCP – Résumé

## B. Configuration DNS

Dans cette partie est présentée la configuration du serveur DNS qui a servi aux manipulations de ce rapport. Il est donc question des paramètres des zones directe et inverse.

### A. Forward zone

Une fois le choix d'ajouter une zone directe fait, nous arrivons sur le lancement de sa configuration comme le montre l'image suivante :



Figure 89 : Configuration DNS - Welcome

Vient ensuite la sélection du type de la zone que nous voulons mettre ensemble. Dans notre cas, nous commençons par sélectionner une zone directe. La configuration de la zone indirecte est présentée après :

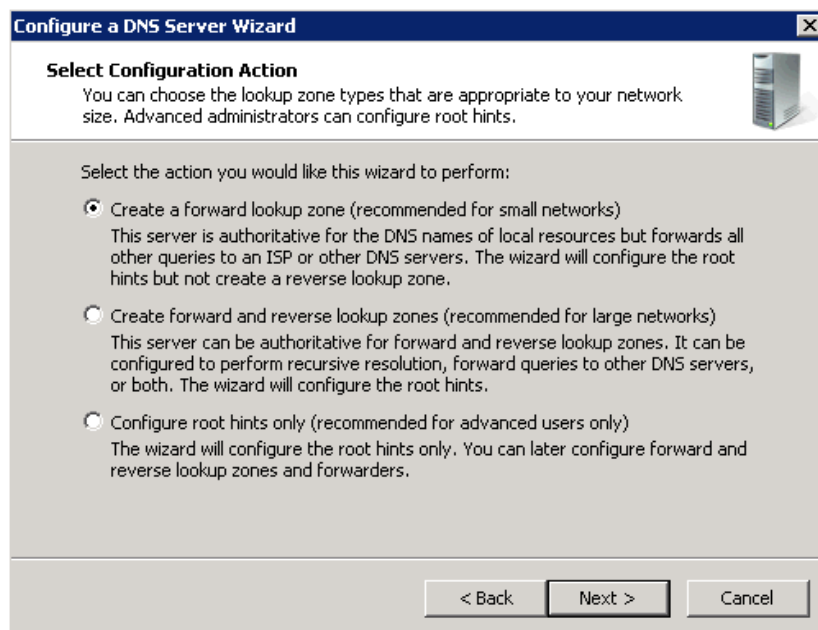


Figure 90 : Configuration DNS – Type de zone

Il est maintenant demandé si c'est le serveur courant qui fait autorité dans la zone. Dans notre configuration, c'est le cas. Cependant, le choix est laissé au lecteur de sélectionner l'option correcte pour son propre type de réseau. Pour ceci, il est invité à se référer à de la documentation qui présente cela.

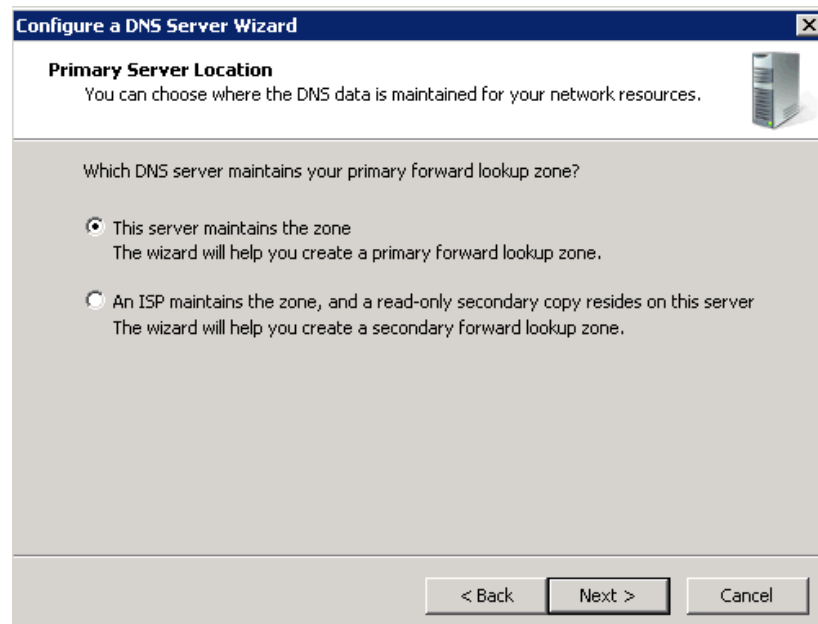


Figure 91 : Configuration DNS – Serveur primaire

Ensuite, il est demandé le nom de la zone que nous voulons créer :

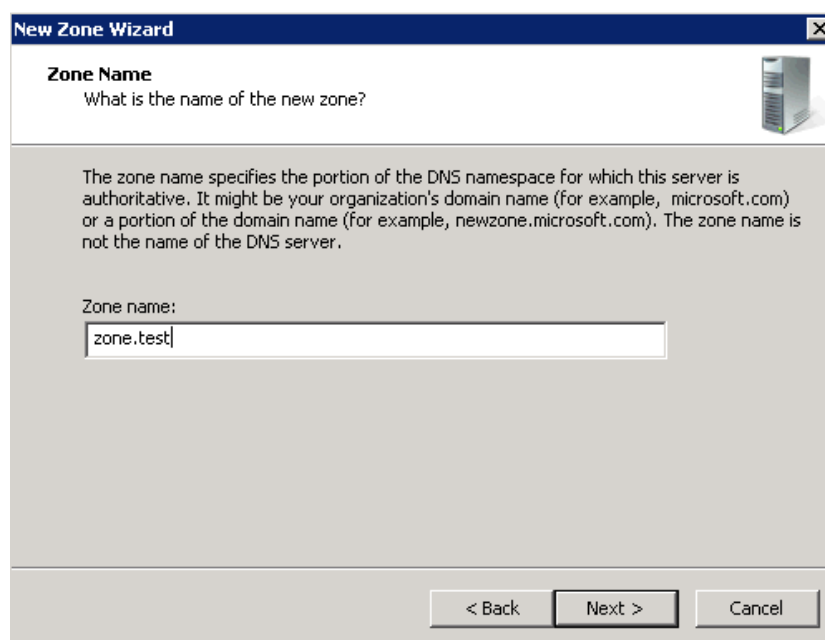


Figure 92 : Configuration DNS – Nom

Puis, comme présenté précédemment (cf. Figure 55), nous pouvons choisir entre différents types de mises à jours. Pour de plus amples informations concernant ce paramètre, veuillez se référer à la partie du rapport s'intitulant : « Enregistrement dynamique ».

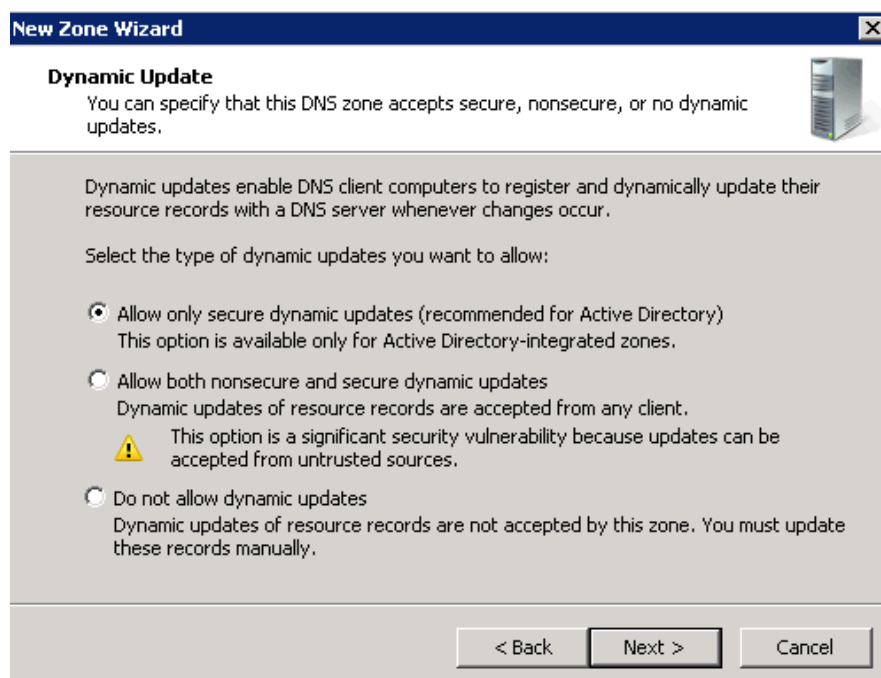


Figure 93 : Configuration DNS – Dynamic updates

Dans le cas où on voudrait transmettre les requêtes DNS auxquelles un serveur DNS ne pourrait pas répondre, nous pouvons ajouter d'autres serveurs. Cependant, cela n'a pas été utile dans le cadre de ce travail. De plus, il

n'y a aucune différence avec ce qui était proposé en IPv4.

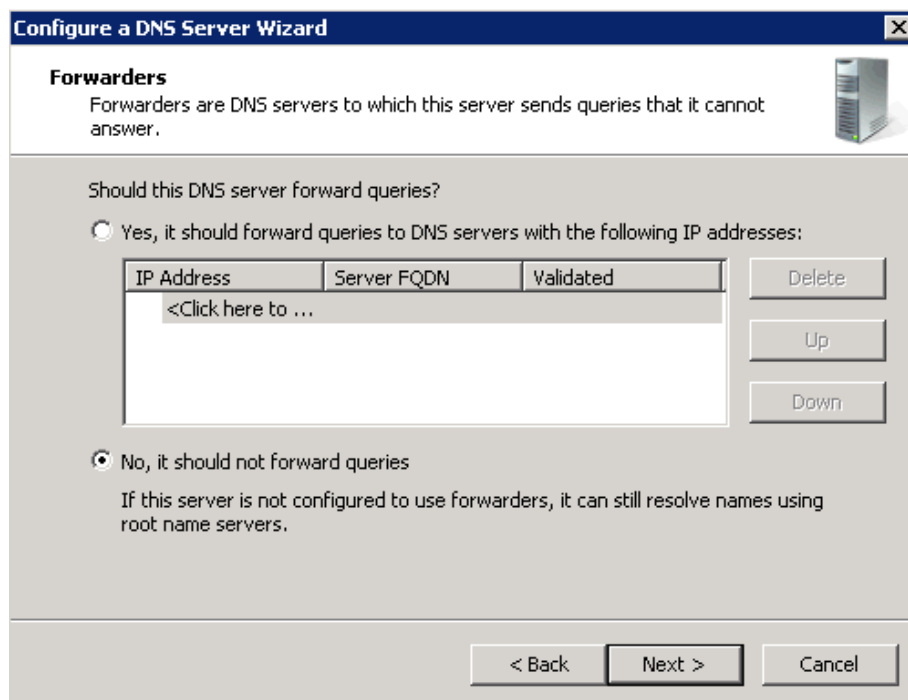


Figure 94 : Configuration DNS - Forwarders

Finalement, tout comme pour les autres configurations, nous arrivons sur une page de résumé qui nous montre les paramètres choisis. Nous pouvons les valider grâce au bouton « Finish ».

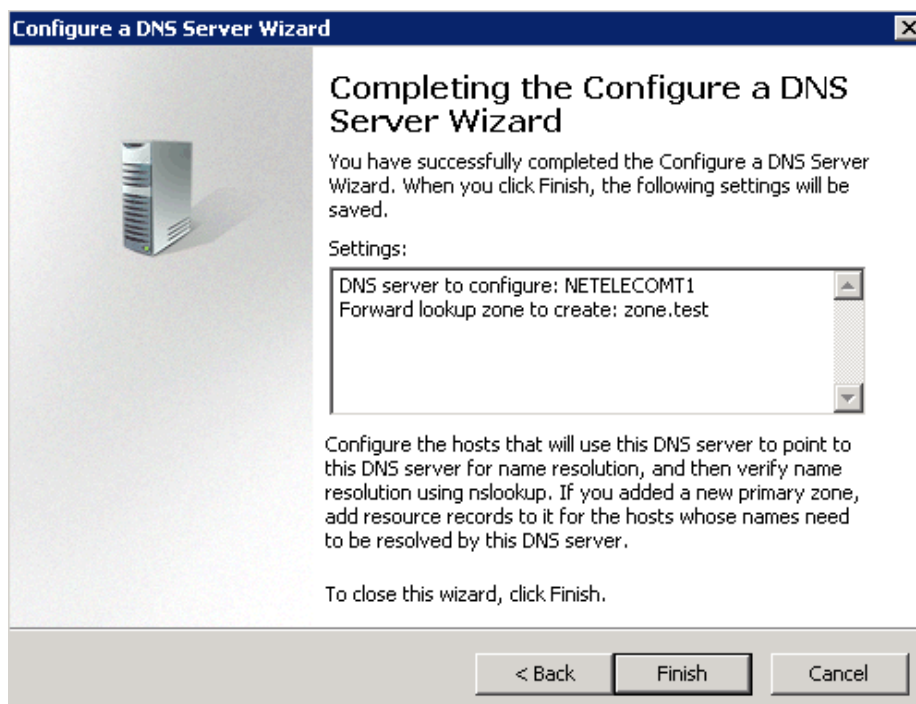


Figure 95 : Configuration DNS - Résumé

Ensuite, dans l'onglet « DNS – Forward Lookup Zones », on peut voir que notre nouvelle zone est apparue.

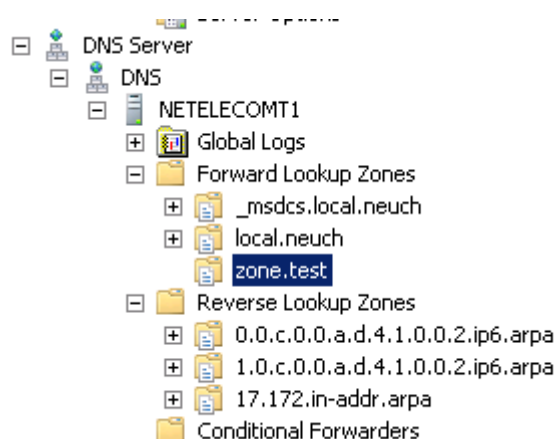


Figure 96 : Configuration DNS – Vérification zone directe

## B. Reverse zone

Voici maintenant la configuration de la zone inverse, qui pour rappel, permet de traduire une adresse IP en un nom de domaine. La première chose à faire est de sélectionner « New Zone » dans l'onglet « Reverse Lookup Zones » du serveur DNS :

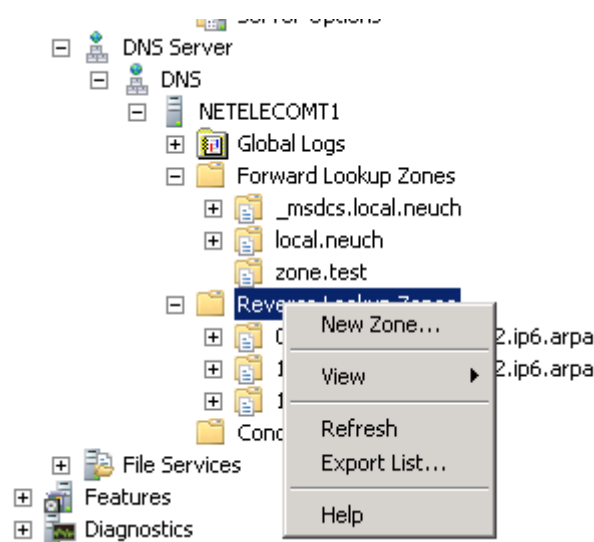


Figure 97 : Configuration DNS – nouvelle zone inverse

Une fois la nouvelle zone sélectionnée, nous arrivons comme pour les autres configurations sur l'écran d'accueil suivant :



Figure 98 : Configuration DNS – Welcome zone inverse

Tout comme pour la zone directe, il nous est demandé si le type de la zone :

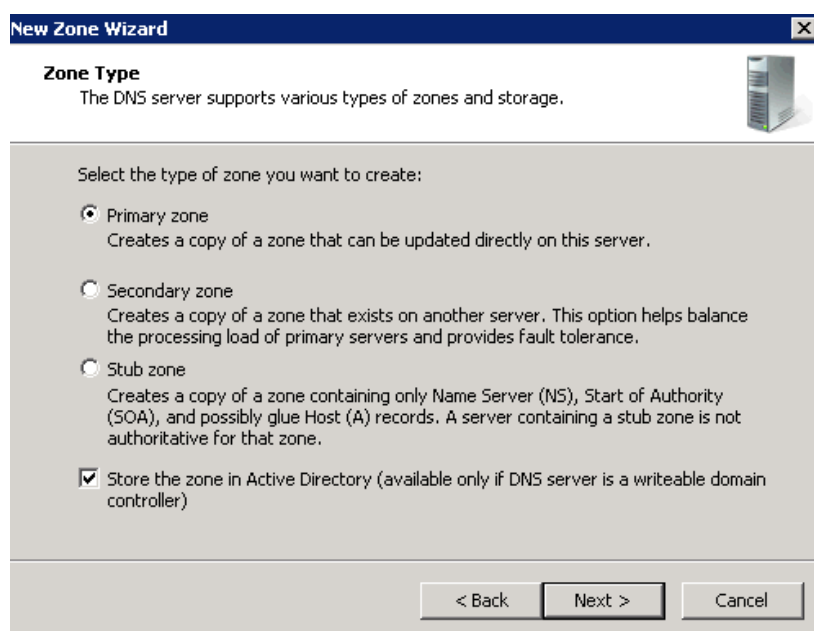


Figure 99 : Configuration DNS – Type de la zone inverse

Puis, il faut choisir si tous les serveurs DNS du « Domain Controller » font partie du domaine, ainsi que le choix de la version d'IP. Dans notre cas, nous choisissons bien sûr IPv6.

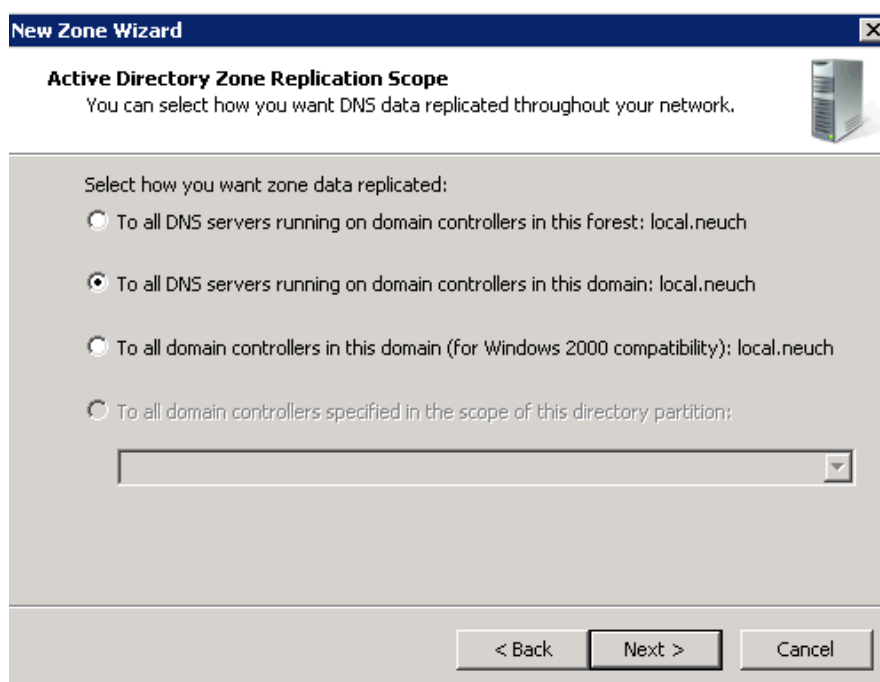


Figure 100 : Configuration DNS – Active directory

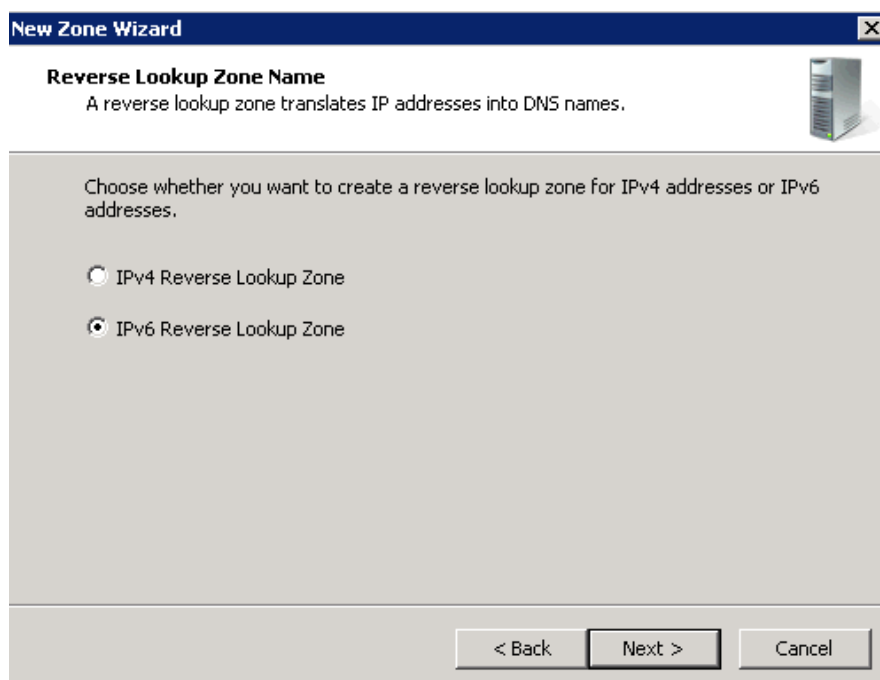


Figure 101 : Configuration DNS – Choix de la version IP

Il reste à définir le préfixe de la zone inverse et la sécurité à appliquer sur les mises à jour dynamiques :



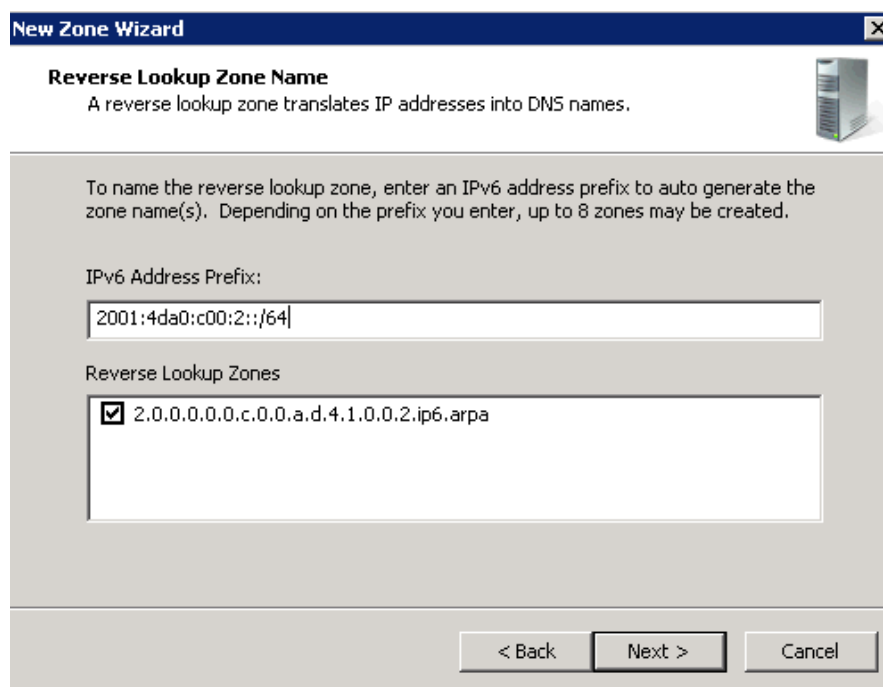


Figure 102 : Configuration DNS – Préfixe de la zone

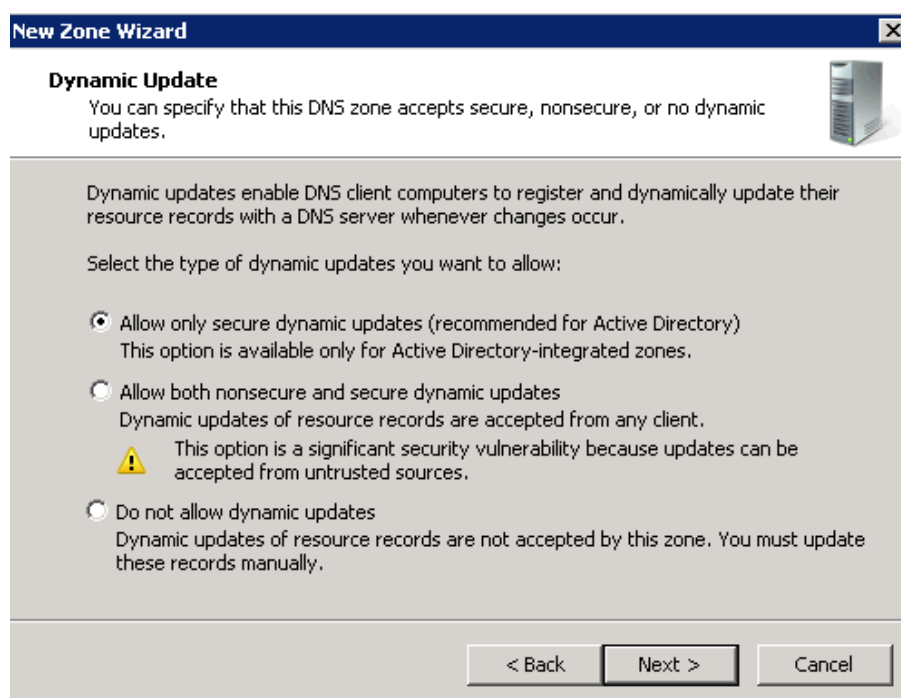


Figure 103 : Configuration DNS – Sécurité mise à jour

Enfin, le résumé des paramètres choisis pour la zone inversée :

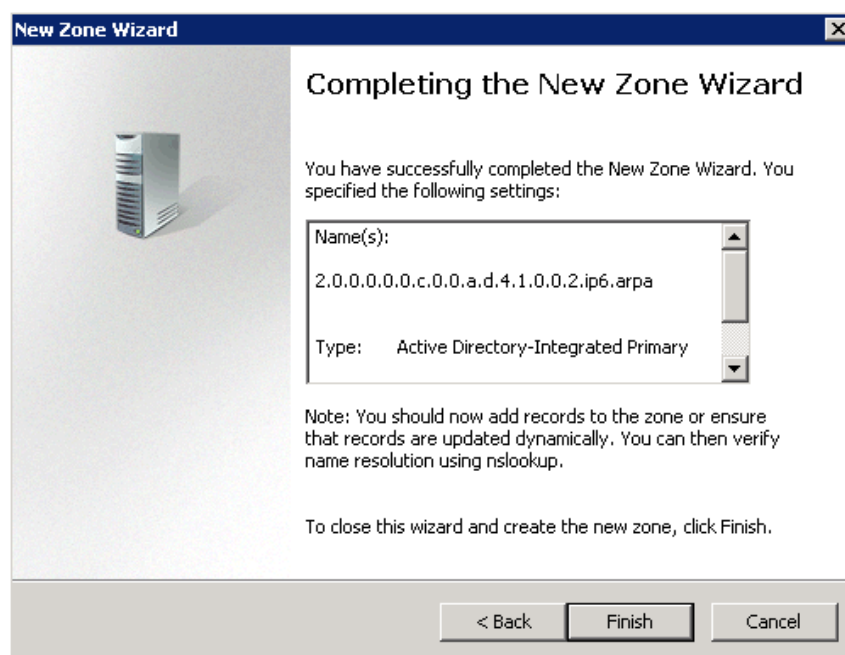


Figure 104 : Configuration DNS – Résumé zone inverse

On peut ensuite voir dans l'onglet « Reverse Lookup Zones » que notre zone a bel et bien été ajoutée correctement :

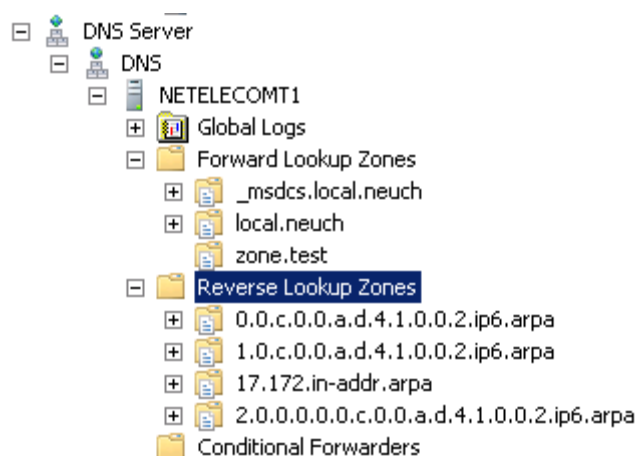


Figure 105 : Configuration DNS – Validation zone inverse

## C. Mise en place d'une infrastructure ASA/AD

Cette partie présente la configuration et l'installation d'une infrastructure permettant de définir des règles par rapports aux utilisateurs sur l'ASA de Cisco. Elle se réfère au schéma de la Figure 82.

Premièrement, les différentes interfaces sont configurées grâce à l'interface graphique de la manière suivante :

Interface	Name	State	Security Level	IP Address	Subnet Mask Prefix Length	Group	Type
Ethernet0/0	Interfac...	Enabled	100	172.17.128.1 2001:4444:5555::1	255.255.255.240 64		Hardware
Ethernet0/1	Interfac...	Enabled	100	192.168.0.1	255.255.255.0		Hardware
Ethernet0/2		Disabled					Hardware
Ethernet0/3		Disabled					Hardware
Management0/0	manage...	Enabled	100	192.168.1.1	255.255.255.0		Hardware/Management Only

Figure 106 : Configuration ASA/AD - Interfaces

A noter que le « security level » a été volontairement été mis à 100 sur toutes les interfaces afin de s'éviter des problèmes de blocage des paquets. La case qui autorise les interfaces ayant la même sécurité à s'échanger des paquets a également été cochée.

La configuration de l'agent, s'effectue grâce aux commandes suivantes, sur l'Active Directory :

### Création du client sur l'AD

```
adacfg client create -name <client-nickname> -ip <IP-address>[/<prefix-length-for-IP-range>] -secret <RADIUS-shared-secret>
```

### Listing des différents clients

```
adacfg client list
```

### Création du domaine contrôleur sur l'agent

```
adacfg dc create -name <DC-nickname> -host <DC-hostname-or-FQDN> -domain <full-DNS-name-of-AD-domain> -user <username-member-of-Domain-Admins-group> -password <password-of-user>
```

### Listing des domaines contrôleur

```
adacfg dc list
```

Une fois ces commandes effectuées, il faut également configurer l'ASA pour le faire communiquer avec l'AD de la manière suivante :

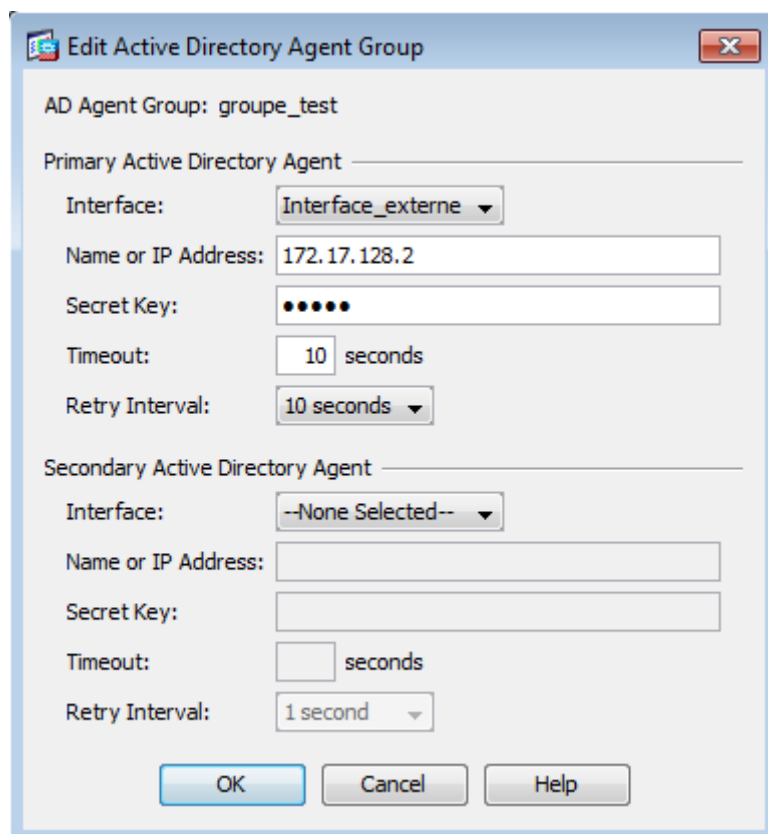


Figure 107 : Configuration ASA/AD – Active Directory Agent Group

A noter que la « secret key » doit être la même que lors de la commande « adacfg client create ».

Ensuite, il est possible de tester si la communication peut être établie en sélectionnant le bouton « test » sur l'interface graphique de l'ASA.

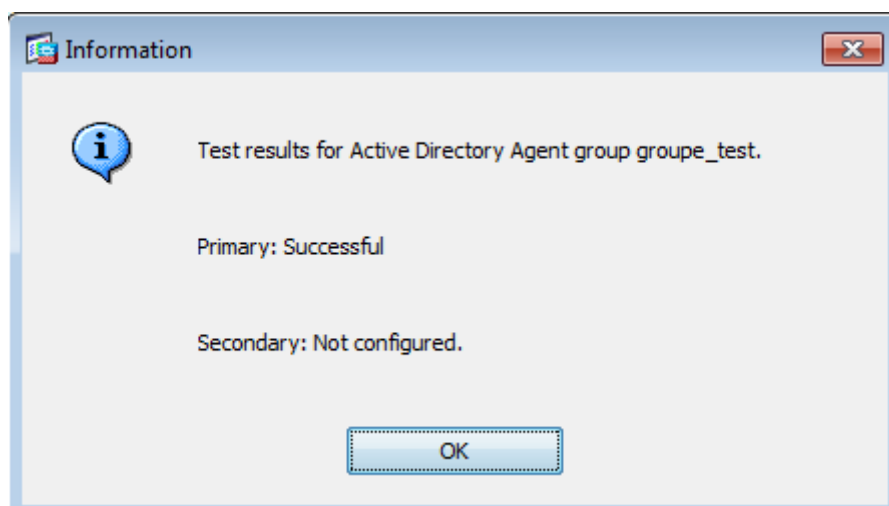


Figure 108 : Configuration ASA/AD - Test AD

Le test doit indiquer « Successful » si la configuration est correcte.

Ensuite, les règles peuvent être définies par rapport aux utilisateurs. Par exemple, si on veut autoriser l'utilisateur « test » faisant partie du domaine « LOCAL » à émettre des pings au travers du firewall, on peut le faire de la manière suivante :










Global (3 rules)							
1	<input checked="" type="checkbox"/>	 192.168.0.2	 any	 icmp	 Permit	TOP 10	3
2	<input checked="" type="checkbox"/>	 any	 LOCAL\test	 any	 icmp	 Permit	TOP 10 14

Figure 109 : Configuration ASA/AD - Règles du firewall

Il ne faut bien sûr pas oublier de définir la règle pour le retour du ping.

Pour une configuration plus précise de l'agent, veuillez se référer au lien suivant :

[http://www.cisco.com/en/US/docs/security/ibf/setup\\_guide/ibf10\\_setup\\_guide.pdf](http://www.cisco.com/en/US/docs/security/ibf/setup_guide/ibf10_setup_guide.pdf)

## D. Installation Direct Access

Les deux sections suivantes présentent la manière dont installer les services requis pour pouvoir configurer un serveur Direct Access fonctionnel. Pour se faire, les points suivants sont obligatoires :

- Une infrastructure PKI permettant la gestion des certificats
- Le service « Web enrollement »
- Deux interfaces physiques (intranet et internet)
- L'interface internet ne doit pas être dans le domaine

Les deux premiers sont des services à installer sur Windows Server 2008, et sont détaillés dans les deux parties suivantes. Les deux derniers sont à vérifier par l'administrateur. Pour le dernier point, il est indispensable que le serveur Direct Access ne soit pas également contrôleur de domaine, sinon toutes ses interfaces feront forcément partie du domaine, et la configuration de Direct Access ne pourra pas se faire. Il est donc préférable d'avoir un serveur supplémentaire faisant office de contrôleur de domaine pour pouvoir mener à bien cette configuration.

### A. Configuration infrastructure PKI

Cette partie présente au lecteur une installation simple d'une infrastructure PKI sur un serveur Windows 2008. Ceci a été mis en place pour pouvoir installer le service « Direct Access ». En effet il est obligatoire d'avoir une infrastructure de ce type pour mettre en place ce service.

En premier lieu, il est nécessaire de choisir le rôle que l'on souhaite installer, qui est dans ce cas « Active Directory Certificate Services ».

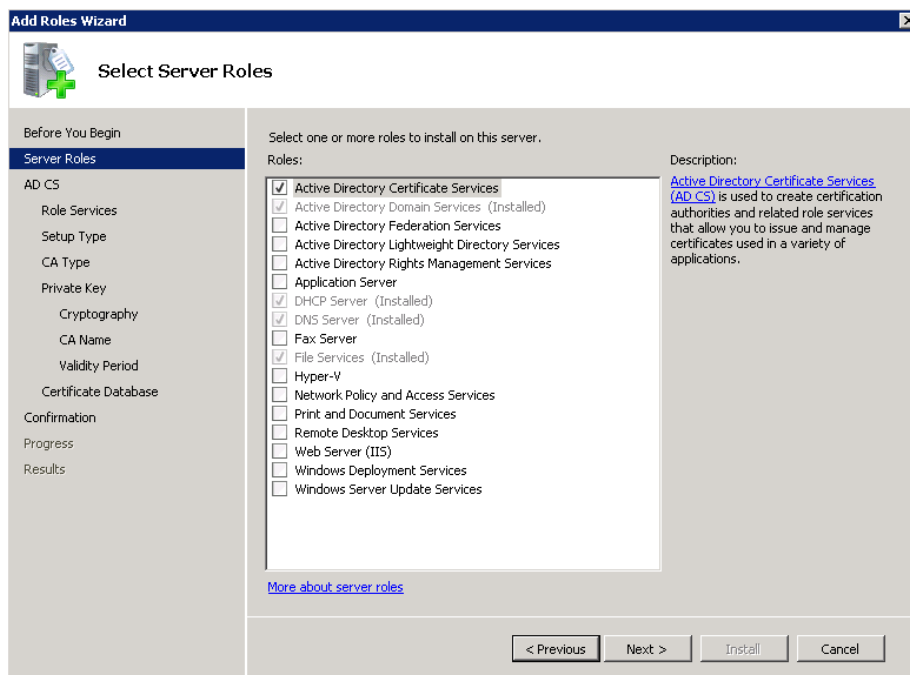


Figure 110 : Configuration infrastructure PKI – Choix du rôle

Vient ensuite le choix du type de CA, que l'on souhaite installer. Sélectionner « Root CA » si le serveur sur lequel est installée la CA est le serveur principal (ou le seul) serveur pouvant délivrer des certificats. Sinon, sélectionner « Subordinate CA ».

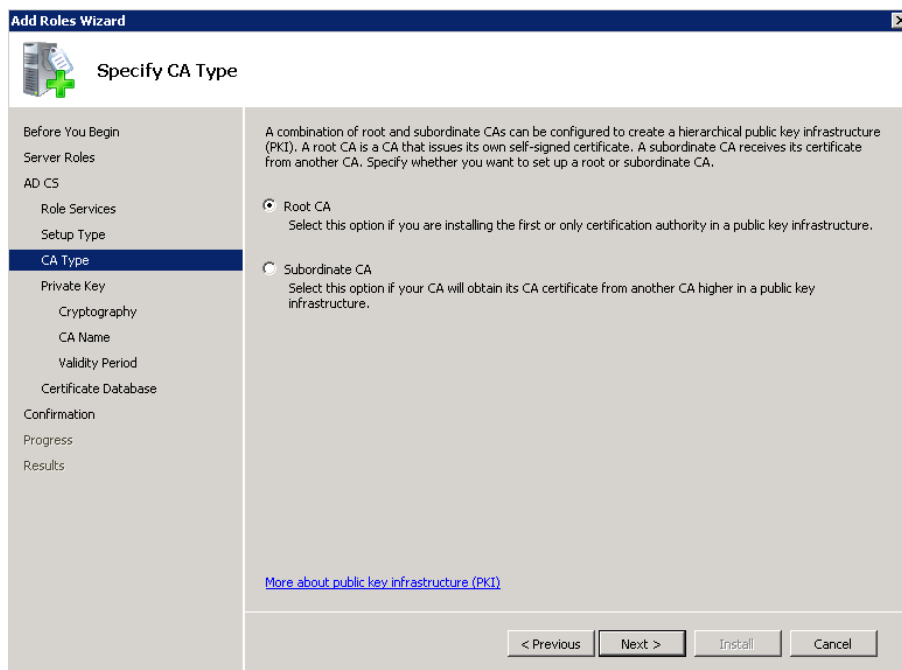


Figure 111 : Configuration infrastructure PKI – Choix du type de CA

Puis, vient la configuration de la clé privée. Dans, le cadre de ce rapport, il a été choisi « Create a new private key » puisqu'aucune clé privée n'existait au préalable. Il est cependant nécessaire d'adapter ce choix si une

infrastructure existe déjà.

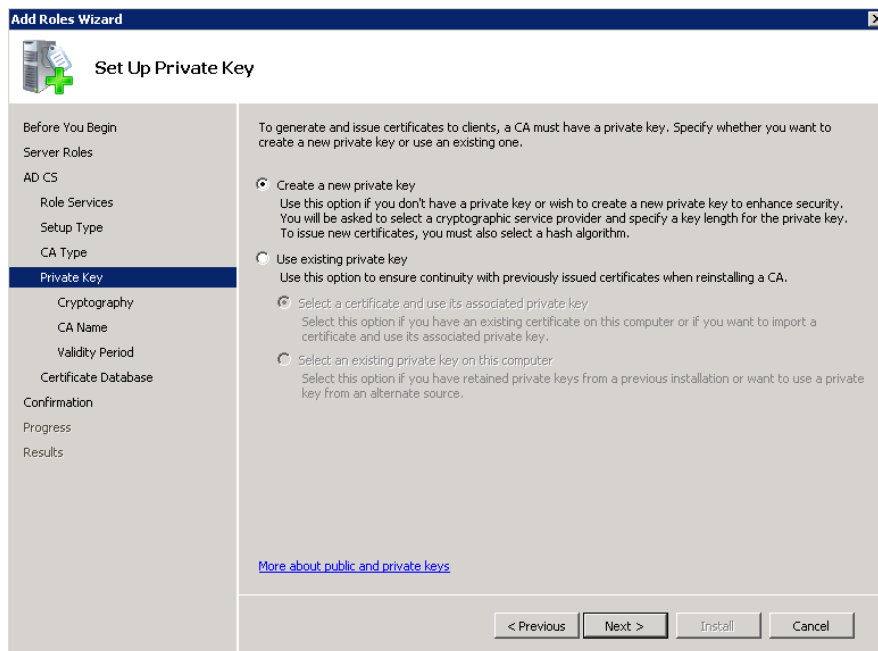


Figure 112 : Configuration infrastructure PKI – Clé privée

Ensuite, il y a la possibilité de choisir l'algorithme de chiffrement à utiliser. Il peut être utile de rappeler que de nos jours, MD5 et SHA-1 ne sont pas inviolables. Il est donc préférable de choisir SHA-512 avec une clé de 2048 caractères. Ceci est bien sûr modifiable en fonction de la criticité des données à sécuriser.

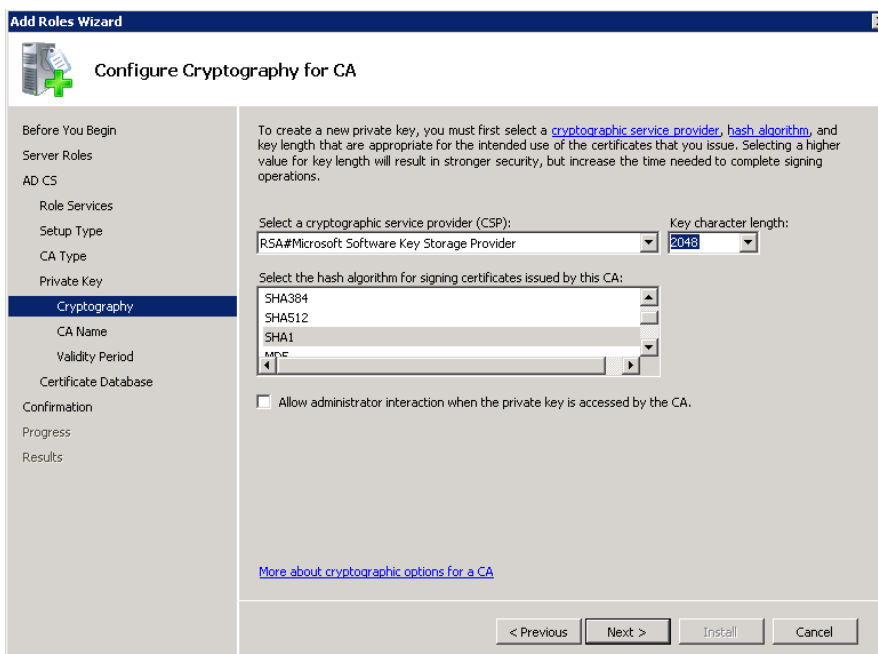
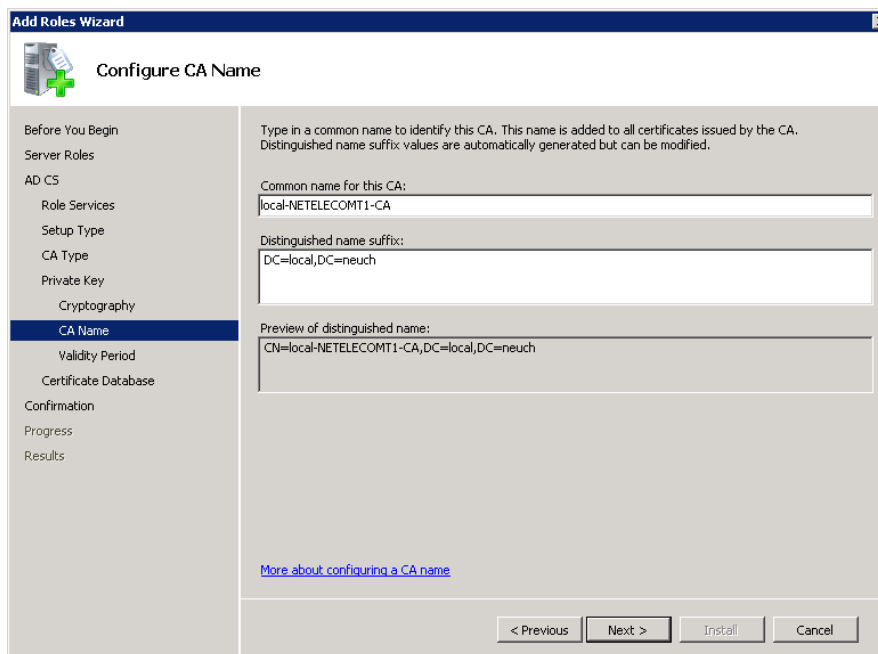


Figure 113 : Configuration infrastructure PKI – Configuration de la cryptographie

Ensuite, il est demandé d'entrer le nom de la CA :





**Add Roles Wizard**

**Configure CA Name**

Before You Begin  
Server Roles  
AD CS  
  Role Services  
  Setup Type  
  CA Type  
  Private Key  
  Cryptography  
  **CA Name**  
  Validity Period  
  Certificate Database  
Confirmation  
Progress  
Results

Type in a common name to identify this CA. This name is added to all certificates issued by the CA. Distinguished name suffix values are automatically generated but can be modified.

Common name for this CA:  
local-NETELECOMT1-CA

Distinguished name suffix:  
DC=local,DC=neuch

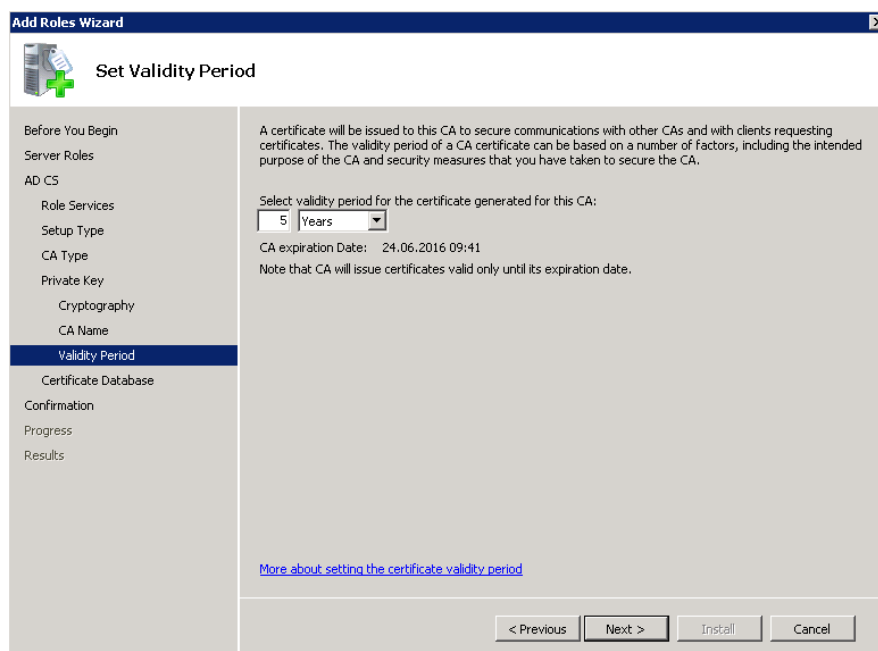
Preview of distinguished name:  
CN=local-NETELECOMT1-CA,DC=local,DC=neuch

[More about configuring a CA name](#)

< Previous   Next >   Install   Cancel

Figure 114 : Configuration infrastructure PKI – Nom de la CA

La période de validité des certificats qui seront délivrés (5 ans par défaut) :



**Add Roles Wizard**

**Set Validity Period**

Before You Begin  
Server Roles  
AD CS  
  Role Services  
  Setup Type  
  CA Type  
  Private Key  
  Cryptography  
  CA Name  
  **Validity Period**  
  Certificate Database  
Confirmation  
Progress  
Results

A certificate will be issued to this CA to secure communications with other CAs and with clients requesting certificates. The validity period of a CA certificate can be based on a number of factors, including the intended purpose of the CA and security measures that you have taken to secure the CA.

Select validity period for the certificate generated for this CA:  
5 Years

CA expiration Date: 24.06.2016 09:41  
Note that CA will issue certificates valid only until its expiration date.

[More about setting the certificate validity period](#)

< Previous   Next >   Install   Cancel

Figure 115 : Configuration infrastructure PKI – Période de validité

Et enfin, le chemin du fichier qui contiendra la base de donnée :

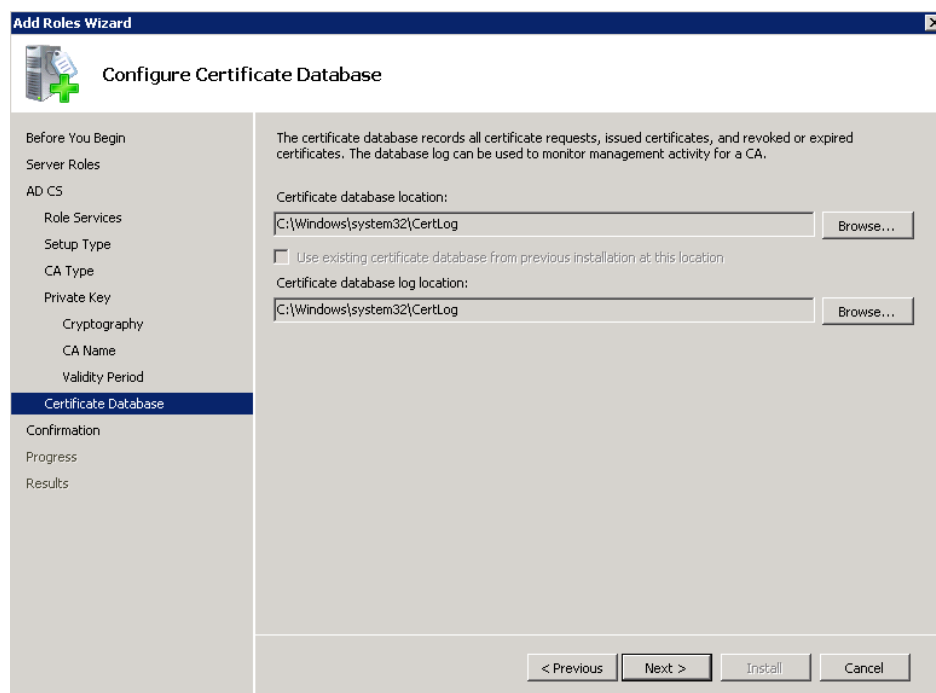


Figure 116 : Configuration infrastructure PKI – Configuration de la base de données

Une fois tous les éléments paramétrés correctement, une fenêtre de confirmation s'affiche qu'il faut valider.

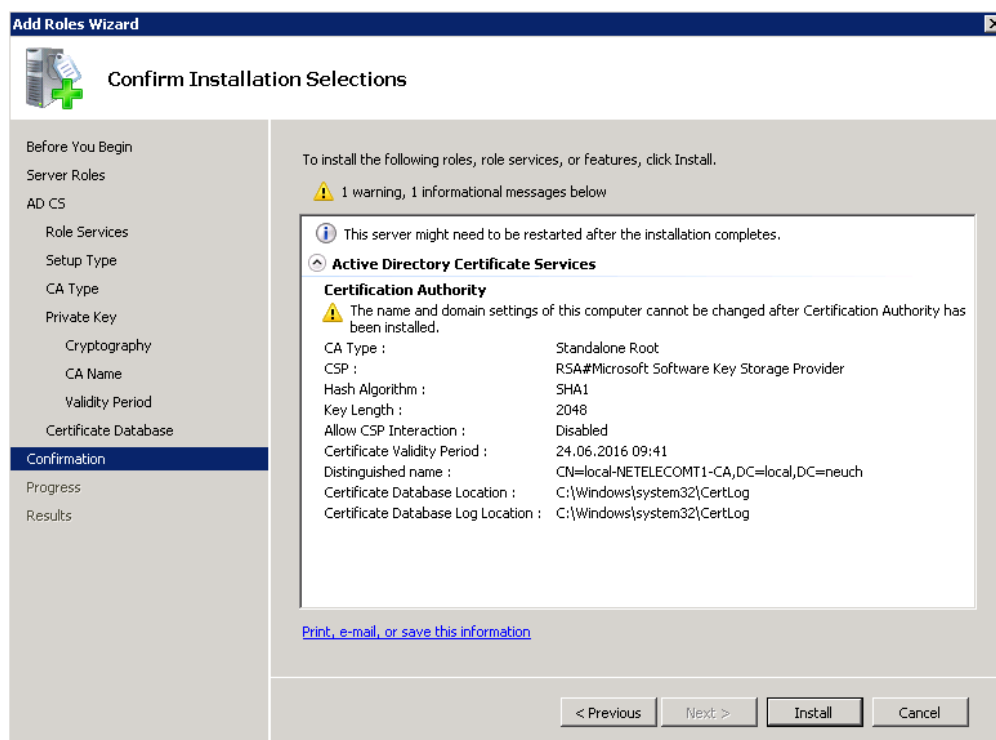


Figure 117 : Configuration infrastructure PKI – Confirmation

L'installation de l' « Active Directory Certificate Services » peut commencer :

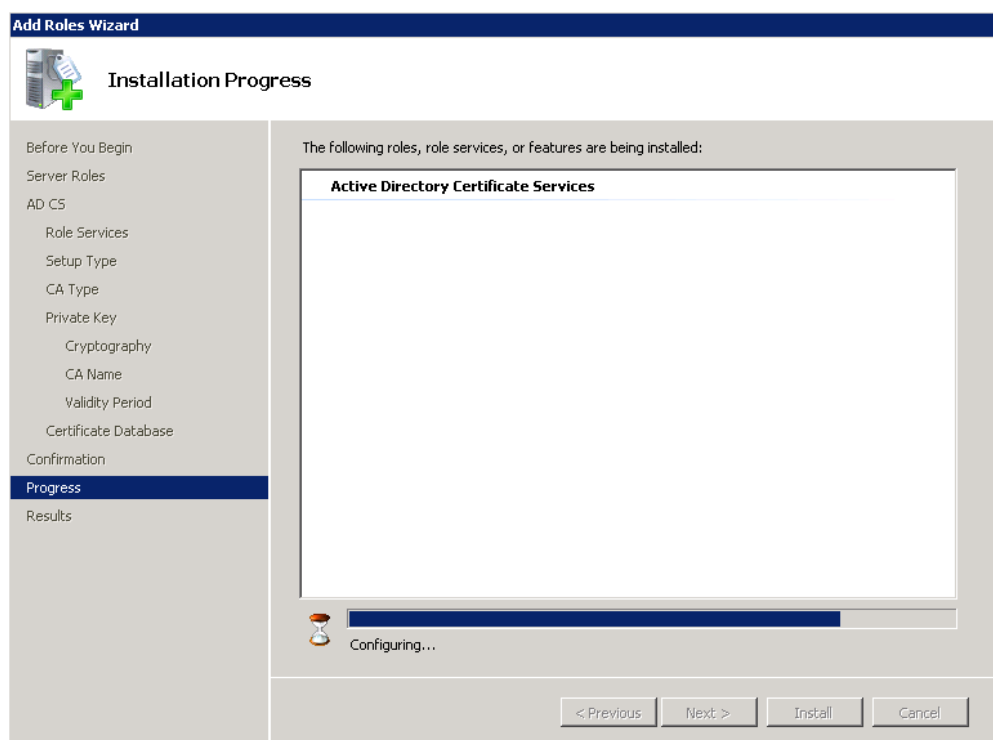


Figure 118 : Configuration infrastructure PKI – Progression de l'installation

Pour terminer, un message de confirmation que l'installation s'est bien passée s'affiche.

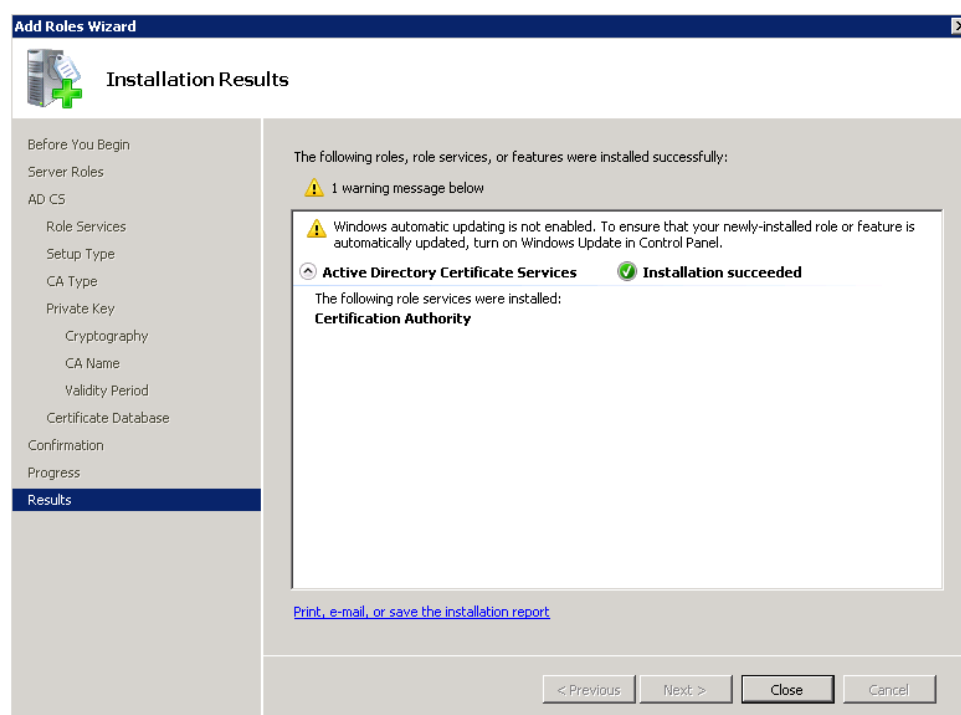


Figure 119 : Configuration infrastructure PKI – Résumé

On peut vérifier que le service se soit bien installé grâce au « Server Manager » de Windows Server.

## E. WEB Enrollement

Cette partie présente la configuration du service « Web Enrollment », nécessaire pour pouvoir ensuite configurer un serveur Direct Access pour un accès VPN sécurisé.

En premier lieu, il faut se rendre dans les « services » à installer, (à l'inverse des installations précédentes où il fallait se rendre dans les rôles), puis choisir « Certification Authority Web Enrollment ».

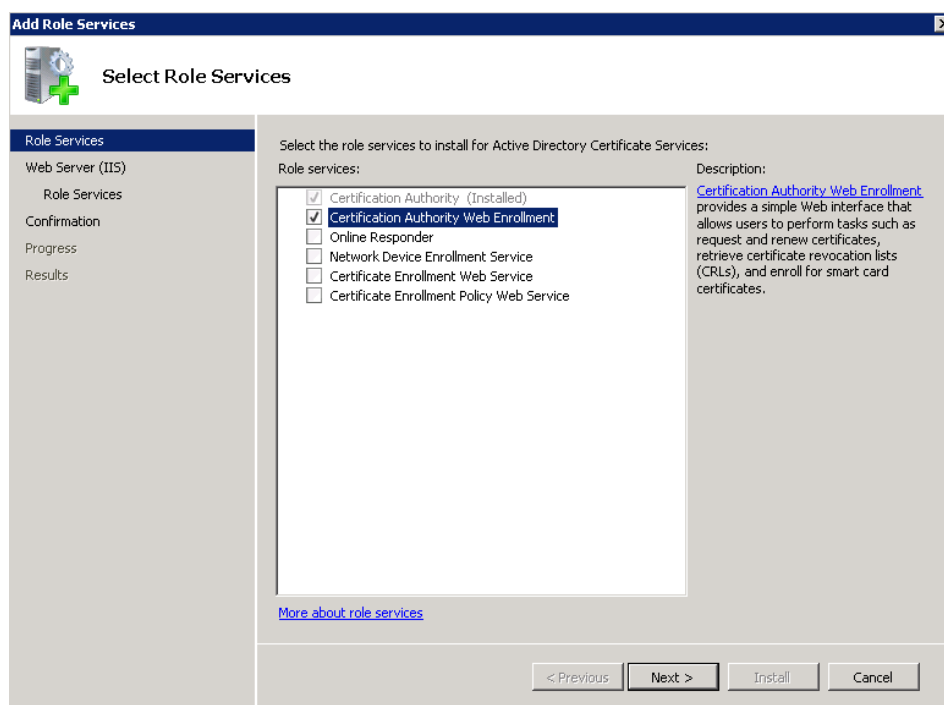


Figure 120 : Configuration Web Enrollment – Choix du rôle

Ensuite, il est demandé de choisir quels services sont à installer sur ce serveur Web. Pour ce travail, le choix a été laissé par défaut.

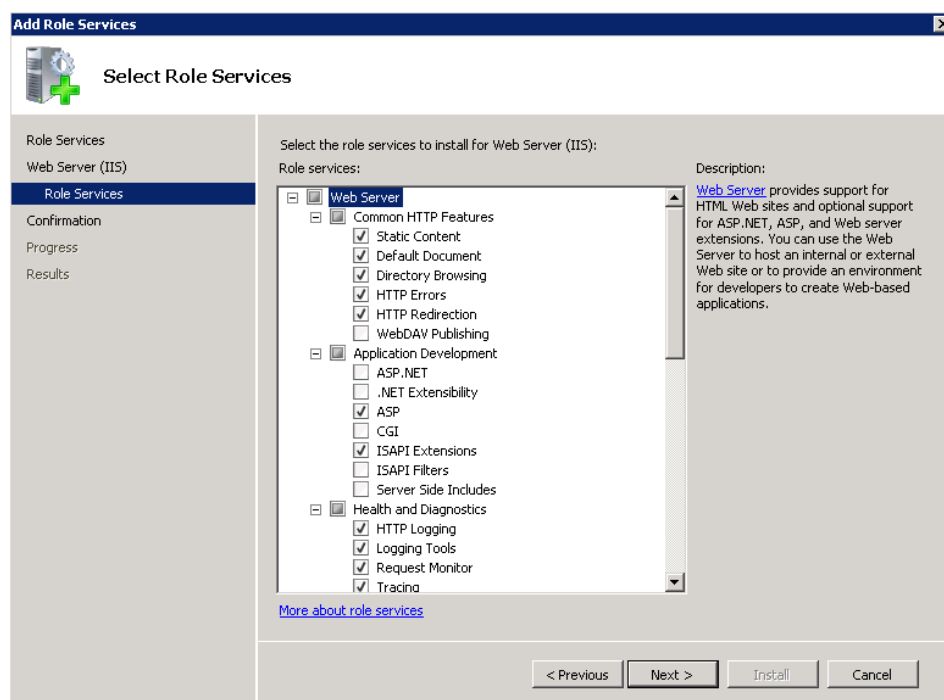


Figure 121 : Configuration Web Enrollment – Choix des caractéristiques à installer

Il suffit ensuite de vérifier les informations entrées concernant l'installation du service, puis de la lancer :

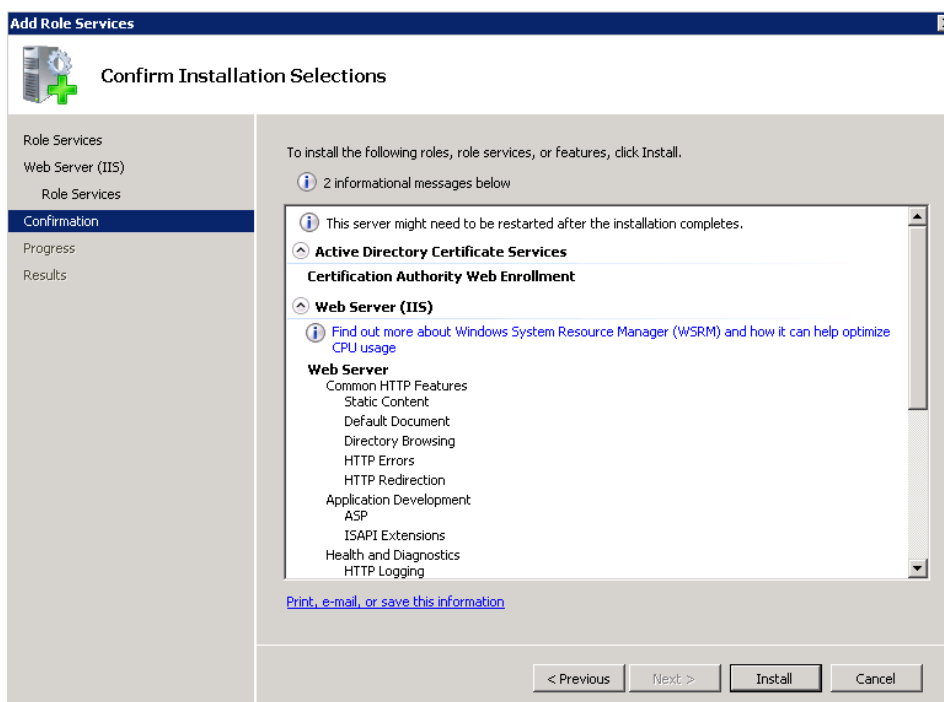


Figure 122 : Configuration Web Enrollment – Confirmation des services à installer

Une fois l'installation terminée, une fenêtre confirme qu'elle s'est terminée avec succès que le service s'est installé correctement.

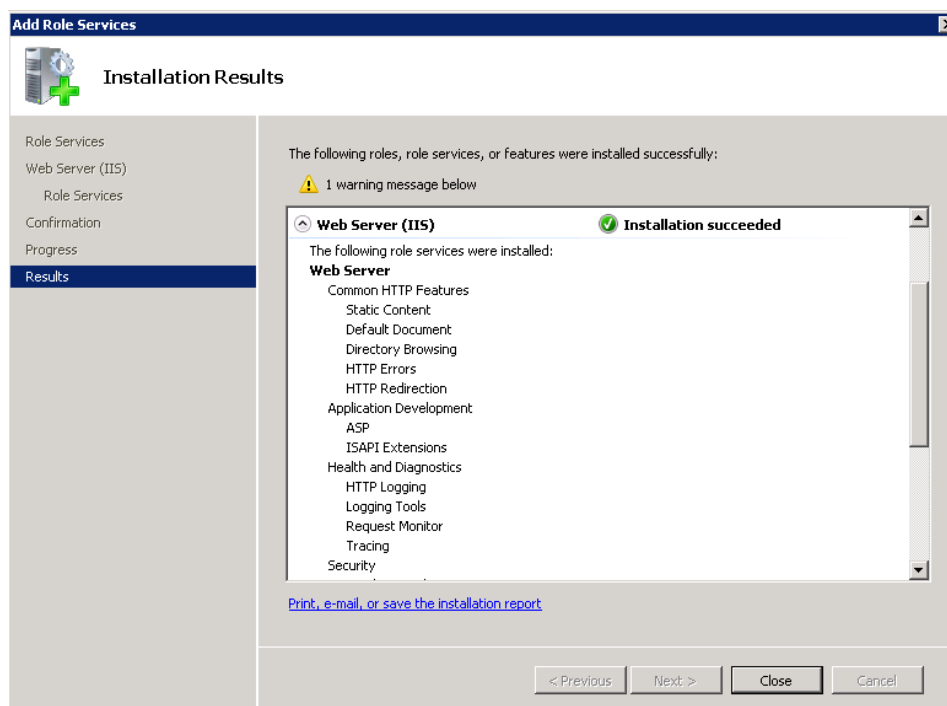


Figure 123 : Configuration Web Enrollment – Résumé

## F. Journal de travail

### Mer 16 février 2011

**9h00 - 9h30** Rencontre avec Jérôme Vernet et découverte des locaux du Service Informatique de Neuchâtel.

**9h30 – 12h00** Présentation par M. Vernet du réseau de Neuchâtel et de son infrastructure.

**13h30 – 15h00** Discussion des principaux objectifs du travail et du cahier des charges.

### Ven 18 février 2011

**9h00 - 12h00** Recherche de documentation sur IPv6, documentation générale sur l'adressage (adresse de lien, adresse général, etc..), lecture de la partie IPv6 du cours de PDR.

**14h00 - 16h00** Recherche et lecture de documentation sur IPv6, documentation générale sur l'adressage, le « Neighbor Discovery », ICMPv6.

**16h00 – 16h10** Début de la rédaction du journal de travail.

### Mer 23 février 2011

**9h00 - 10h00** Présentation par M.Vernet de la salle serveur, de la place de travail et du matériel à disposition.

**10h00 – 12h00** Début de la conception de l'architecture du prototype 1 qui va servir de base pour la première partie de ce travail.

**13h00 – 17h10** Recherche et lecture de la documentation trouvée jusqu'à présent, principalement sur les protocoles DNSv6 et DHCPv6.

## Ven 25 février 2011

**10h00 – 12h00** Lecture de la documentation sur Windows Server 2008.

**13h30 - 16h00** Lecture de la documentation sur la gestion DNS et DHCP sur Windows Server 2008.

**16h00 – 16h10** Rédaction du journal de travail.

## Mer 9 mars 2011

**09h00 – 12h00** Lecture de documentation sur Windows Server 2008 (le serveur n'étant pas totalement prêt).

**13h30 – 17h00** Le serveur étant opérationnel, j'ai pu configurer la fonction DHCP par rapport au réseau que nous avons configuré. Les tests se sont révélés concluants puisque les hôtes ont pu récupérer les adresses configurées.

**17h00 – 17h10** Rédaction du journal de travail.

## Ven 11 mars 2011

**10h00 – 12h00** Rédaction d'un résumé sur les principales fonctionnalités d'IPv6, de ce qui a été lu jusqu'à présent afin d'être plus efficace par la suite lors de la configuration des services.

**13h30 – 15h00** Mise sous version informatique du réseau prototype 1.



**Ven 18 mars 2011**

**09h00-11h00** Installation du serveur DNS pour IPv4, avec comme zone principale : local.neuch.

**11h00-12h00** Test du serveur DNS. Pour se faire, j'ai installé un serveur web sur mon laptop, j'ai inscrit son entrée A dans le DNS et j'ai pu tester s'il était accessible depuis un autre laptop, ce qui était le cas.

**13h00-14h00** Ajout d'une zone de recherche inversée sur le DNS. J'ai également regardé les différentes options disponibles pour la configuration du serveur DNS.

**14h00-16h00** Recherche de documentation sur la façon dont inscrire dynamiquement les clients sur le serveur DNS. Je n'ai malheureusement pas réussi à résoudre ce problème. Il a été conclu avec M.Vernet que nous regarderons ceci à son retour de vacances, puisque cette configuration présente quelques subtilités. Le problème ne préterite par contre pas l'avancée du projet en général.

**Mer 23 mars 2011**

**09h00-10h00** Réflexion sur l'architecture de notre réseau IPv6.

**11h00-12h00** Les clients ayant un problème de lenteur lors de l'acquisition de l'adresse IPv4 vers le dhcp, nous avons cherché à le résoudre. Il s'est avéré que c'est le spanning tree sur les routeurs qui prenaient énormément de temps. Nous avons donc désactivé cette fonction.

**13h00-14h00** Configuration du DHCPv6 sur Windows Server 2008. Je n'ai pas encore pu tester cette fonctionnalité, car il nous a fallu d'abord que le réseau IPv6 fonctionne, ce qui a été le cas à la fin de la journée.

**14h00-15h00** Configuration du DNS version 6 sur le serveur. Même remarque que pour le DHCP.

## Mer 23 mars 2011(fin)

**15h00-16h30** Analyse des paquets envoyés par une station lors d'une configuration stateless de son adresse IPv6 afin de se faire une idée de comment se comporte le « client » et le routeur.

## Mer 06 avril 2011

**09h00 – 12h00** Début de la rédaction du rapport intermédiaire, parties « adressage » et « prototype ».

**13h00-14h00** Tests effectués avec Julien concernant la configuration du DHCPv6. Ces tests nous ont permis de mettre en place le DHCPv6 correctement. Les hôtes, peu importe leur réseau, peuvent maintenant acquérir une adresse du DHCP.

**14h00 – 15h00** Tests du DNS IPv6. Le DNS fonctionne correctement en IPv6, zone de recherche normale comme inversée. Cependant, le problème de la mise à jour automatique des enregistrements par les clients persiste. Il est donc nécessaire pour l'instant d'enregistrer manuellement les clients dans le DNS.

**15h00-16h15** Fin des tests et rédaction du rapport intermédiaire + journal de travail.

**Mer 13 avril 2011**

**09h00 – 12h00** Recherche de documentation concernant le problème de l'enregistrement dynamique des clients sur le DNS et installation d'un active directory.

***Remarque :** L'installation de l'Active Directory a permis de résoudre tous les problèmes concernant l'enregistrement dynamique des clients. En effet, le serveur doit faire partie d'un domaine pour que le dns soit mis à jour.*

**13h00 – 16h00** Rencontre avec M.Robert.

**16h30 – 17h00** Rédaction du journal de travail et rangement.

**Mer 20 avril 2011**

**09h00 – 12h00** Il reste un problème concernant l'enregistrement des clients dans le DNS en zone inversée. J'ai donc cherché la cause du problème et l'ai résolu. Il venait du fait qu'une sécurité était mal configurée au niveau du serveur DNS. Du coup, le serveur refusait de mettre à jour la zone inversée.

Rédaction d'une partie du rapport.

**13h00-16h30** Tests avec Julien de différents cas de figure de configuration d'adresse, notamment la problématique du stateless-statefull. Ces tests ont été effectués essentiellement pour notre compréhension personnelle, et d'autres vont suivre qui seront détaillés par des captures d'écran dans le rapport.

**16h30-17h00** Rédaction du journal de travail et rangement.

**Ven 6 mai 2011**

**09h00 – 12h00** Recherche de documentation sur les « scope options » du DHCPv6 et rédaction de ceci dans le rapport.

Prise de capture d'écran concernant les configurations des différents serveurs installés tout au long du travail : DNS, DHCP, AD.

**13h00 – 14h00** Tests des différents niveaux de sécurité au niveau des mises à jours dynamique des clients sur le DNS. Celles-ci fonctionnent correctement en fonction de si le client se trouve dans le domaine ou pas et de la sécurité choisie.

**14h00 – 15h00** Après avoir désactivé les services de l'AD, j'ai rencontré un problème de login sur le serveur. Ceci venait du fait que le service d'authentification du serveur a été désactivé par erreur. Il a donc fallu redémarrer le serveur pour résoudre ce problème.

**15h00 – 16h30** Réflexion et rédaction des différents tests à effectuer pour bien expliquer les différences entre les modes du DHCP (stateless, statefull, etc...) ainsi que les requêtes DNS IPv6, le fonctionnement du protocole ARP, le choix du stack IP dans le cas d'un dual stack, etc...

Le but est que la prochaine fois, ces tests soient effectués de manière détaillée avec des captures d'écran et Wireshark à l'appui pour pouvoir documenter cette partie dans le rapport.

**16h30 – 17h00** Rédaction du journal de travail et rangement.

**Mer 11 mai 2011**

**09h00 – 12h00** Tests des scénarios écrits lors de la dernière séance. Ces scénarios sont en rapport avec DHCPv6, DNS et ICMP et figureront dans le rapport final afin d'expliquer en détails les mécanismes mis en œuvre dans IPv6 grâce à des captures Wireshark.

**13h00 – 16h30** Idem.

**16h30 – 17h00** Début de la rédaction du rapport.

**Mer 18 mai 2011**

**09h00 - 12h00** Capture Wireshark des différents scénarios afin d'expliquer les différentes utilités du protocole Neighbor Discovery (NUD, Adresse Dupliquée, Traduction d'adresse, etc...).

**13h00 – 16h30** Rencontre avec M. Robert et discussion sur les points à améliorer et à faire, notamment par rapport au rendu du travail intermédiaire.

**16h30 – 17h00** Rédaction du journal de travail.

## Mer 25 mai 2011

**09h00 - 12h00** Rédaction du rapport en fonction de ce qui a été fait le 18 mai : Protocole Neighbor Discovery + Types d'adresse IPv6 et début du canevas général du rapport.

**13h00 – 16h30** Captures finales pour les explications concernant les protocoles DHCPv6 et DNS. Ces captures seront intégrées au rapport afin d'expliquer ces deux protocoles précisément.

**16h30 – 17h00** Rédaction du journal de travail.

## Mer 1<sup>er</sup> Juin 2011 (Yverdon)

**Toute la journée** Cette séance a été entièrement consacrée à la rédaction du rapport, notamment les parties traitant sur le DHCP et le DNS.

## Mer 8 Juin 2011

**09h00 - 12h00** Réalisation des dernières captures Wireshark nécessaires à la réalisation du rapport, car certaines étaient incomplètes ou erronées.

**13h00 – 16h30** Rédaction du rapport.

**16h30 – 17h00** Rédaction du journal de travail.

## **Mer 15 Juin 2011**

**09h00 – 12h00** Mise en page du rapport : page de titre, table des matières, titres, en-têtes et pieds de pages, mise en page générale.

**13h00 -16h30** Schémas Visio pour la présentation des différents scénarios.

**16h30 – 17h00** Rédaction du journal de travail.

## **Jeudi 16 Juin 2011**

**12h00 – 15h00** Rédaction de l'introduction, fin de la mise en page.

## **Vendredi 17 Juin 2011**

**12h00-15h00** Correction des erreurs, lecture finale et envoi.

## lundi 20 juin 2011

**08h45 - 10h00** Discussion avec Jérôme sur le prototype 2, vue d'ensemble des protocoles à mettre en place + équipements.

**10h00 - 11h30** Réflexion sur le nouveau prototype2, établissement du schéma, définition des VRFs.

**12h30 - 16h30** Recherche et lecture de documentation sur MPLS, VRFs, OSPF (types d'area).

**16h30 - 17h00** Rédaction du journal de travail et rangements.

## mardi 21 juin 2011

**08h40 - 10h00** Installation du toolkits thc-ipv6 (Linux Fedora 14) afin de pouvoir tester différentes attaques sur le protocole IPv6.

**10h00 - 10h45** Montage d'un petit réseau avec un switch + une machine attaquante + une machine cible en IPv6 pour tester les attaques.

**10h45 - 11h45** Lecture des « man pages » du toolkit afin de voir comment cela fonctionne.

**12h45 - 15h00** Tests des différentes attaques : Scanning, MITM, Fuzz, Dos, flooding réseau + observations.

**15h00 - 16h30** Recherche et lecture de doc sur le protocole SEND (SEcure Neighbor Discovery) qui est censé servir à contrer les attaques testées.

**16h30 - 17h00** Rédaction du journal de travail et rangements.



## mercredi 22 juin 2011

**08h45 - 12h00** Installation d'un serveur DHCP sur Linux et Windows mais sans succès... Le serveur ne répond pas aux requêtes du client => à voir avec Jérôme.

**13h00 - 14h00** Installation d'un serveur DHCP IPv4 et IPv6 sur un routeur Cisco. Cette fois-ci tout fonctionne bien. Je dois maintenant attendre que le prototype 2 soit complètement opérationnel pour tester une attaque de type "DHCP snooping".

**14h00 - 16h30** Rencontre avec M.Robert afin de discuter du rapport intermédiaire et de la suite des opérations.

**16h30 - 17h00** Rédaction du journal de travail et rangements.

## jeudi 23 juin 2011

**08h45 - 12h00** Recherche et lecture de documentation sur Microsoft Direct Access et du service de "Certificate Authority" pour la gestion d'une infrastructure PKI (nécessaire à Direct Access).

**12h00 - 16h30** Installation de Direct Access.

### *Problèmes rencontrés :*

- Nécessite deux interfaces physiques (Une demande à l'administrateur système a été faite pour en avoir une supplémentaire).
- Configuration complexe de l'infrastructure PKI.
- Serveur WEB ISS obligatoire.

**16h30 - 17h00** Rédaction du journal de travail et rangements.

**vendredi 24 juin 2011**

**08h45 - 12h00** Lecture de documentation sur Microsoft Direct Access, suite de l'installation.

**14h00 - 17h30 (à la maison)** Retour sur les points du rapport intermédiaire discutés lors de la séance de Mercredi.

**lundi 27 juin 2011**

**8h45 - 12h00** Résolution du problème des deux interfaces physiques. Réinstallation de la PKI (qui n'était pas configurée correctement).

**13h00 - 14h00** Etablissement du plan d'adressage IPv6 du prototype 2 : Prefixe par VRF et ensuite par sous-réseau à l'intérieur de ces VRFs.

**14h00 - 16h30** Fin de l'installation de Direct Access.

***Problème :***

- L'interface reliée à internet ne doit pas être dans le domaine.

***Solution :***

- Il est impossible d'avoir un serveur qui fait à la fois office de "Contrôleur de domaine" et de "Direct Access Server" en même temps (car les deux interfaces d'un contrôleur de domaine font forcément partie du domaine). L'installation n'a pas pu se faire complètement puisqu'il n'y avait pas deux serveurs disponibles pour séparer les deux services.

**16h30 - 17h00** Rédaction du journal de travail et rangements.

## mardi 28 juin 2011

**09h00 - 12h00 (maison)** Correction du rapport selon les discussions faites lors de la dernière séance.

**13h00 - 17h30** Séminaire IPv6 au SIG à Genève.

## mercredi 29 juin 2011

**08h45 - 12h00** Captures d'écran de toute la configuration de PKI + DirectAccess en vue de les mettre dans le rapport.

**13h00 - 16h30** Début de la rédaction sur la configuration de Direct Access (en annexe) + début de la rédaction de la partie sécurité du rapport.

**16h30 - 17h00** Rédaction du journal de travail et rangements.

## jeudi 30 juin 2011

**08h45 - 10h00** Présentation du Firewall ASA de Cisco par Jérôme en vue de tester les fonctionnalités IPv6 compatibles avec celui-ci.

**11h00 - 12h00** Lecture de documentation sur le Firewall ASA.

**13h00 - 16h30** Elaboration d'un petit schéma de test pour établir un VPN + Montage et configuration des éléments (1 routeur, 1 PC, 1 Cisco ASA) + Configuration du Cisco ASA.

**16h30 - 17h00** Rédaction du journal de travail et rangements.

## Vendredi 1<sup>er</sup> Juillet 2011

**08h45 – 12h00** Configuration du VPN IPsec en IPv4 à l'aide du Cisco ASA. L'idée est ensuite de monter un VPN IPsec IPv6.

**13h00 – 16h30** Tests du VPN en IPv4. Malheureusement, le Cisco ASA n'offre pas de fonctionnalités IPv6 pour faire des VPN mais seulement une compatibilité au niveau Firewall. Cette incompatibilité sera mentionnée dans le rapport.

## lundi 4 juillet 2011

**08h45 - 12h00** Rédaction de la partie annexe concernant l'installation d'une CA sur Windows Server 2008.

**13h00 - 16h45** Début de la rédaction de la partie sécurité + captures Wireshark pour illustrer cette partie.

**16h45 - 17h00** Rédaction du journal de travail.

## mardi 5 juillet 2011

**08h45 - 12h00** Rédaction de la partie sécurité du rapport.

**13h00 - 16h45** Rédaction de la partie sécurité du rapport.

**16h45 - 17h00** Rédaction du journal de travail.

## mercredi 6 juillet 2011

**08h45 - 12h00** Reformulation de certaines phrases de la partie DHCP + rédaction de l'introduction de cette partie.

**13h00 - 15h00** Reformulation de certaines phrases de la partie DNS + rédaction de l'introduction de cette partie.

**15h00 - 16h30** Début de la rédaction de la partie Direct Access.

**16h30 - 17h00** Rédaction du journal de travail et rangements.

## Jeudi 7 juillet 2011

**08h45 – 12h00** Rédaction de la partie « Direct Access » du rapport.

**13h00 – 17h30** Rédaction de la partie « Direct Access du rapport.

## Vendredi 8 juillet 2011

**08h45 – 12h00** Correction des erreurs, orthographe, syntaxe et reformulation de certaines phrases de la première partie du rapport.

**13h00 – 17h30** Rédaction de la partie « SEND » du rapport.

## Dimanche 10 juillet 2011

**09h00 – 12h00** Intégration du journal de travail dans le rapport final, mise en page de la deuxième partie du rapport.

**14h30 – 18h00** Rédaction du glossaire + Rédaction de la partie annexe concernant l'installation du service « Web Enrollement » nécessaire à l'installation de « Direct Access » sur Windows Server 2008.

## Lundi 11 juillet 2011

**09h00 – 12h00** Rencontre avec M.Robert

**13h00 – 16h45** Tests sur le DHCP spoofing. J'ai remarqué que les commandes utilisées en IPv4 n'existent pas en IPv6. La solution pour la nouvelle norme est donc d'utiliser des « access list » par port. Ceci sera détaillé dans la partie « DHCP spoofing » du rapport. Rédaction de la partie « DHCP spoofing » du rapport.

**16h45 – 17h00** Rédaction du journal de travail et rangements.

## Mardi 12 juillet 2011

**09h00 – 12h00** Mesure de performance sur les routeurs avec Julien + Rédaction de la partie « adresses dupliquées ».

**13h00 - 16h45** Suite de la rédaction de la partie « Direct Access » du rapport.

**16h45 – 17h00** Rédaction du journal de travail.

## Mercredi 13 juillet 2011

**09h00 – 12h00** Rédaction du rapport, début de la partie « Cisco ASA ».

**13h00 - 16h45** Discussion avec Jérôme de ce qu'il reste à faire concernant l'ASA, notamment le plugin à installer sur l'AD + Installation du plugin.

**16h45 – 17h00** Rédaction du journal de travail et rangements.

## Jeudi 14 juillet 2011 (Yverdon)

**09h00 – 12h00** Rédaction du rapport, relecture + correction des erreurs

**13h00 - 16h45** Rédaction du rapport, relecture + correction des erreurs

**16h45 – 17h00** Rédaction du journal de travail.

## Vendredi 15 juillet 2011

**09h00 – 12h00** Configuration du plugin de l'AD le liant à l'ASA. Recherche et lecture de documentation à ce sujet.

**13h00 - 16h30** Configuration de l'ASA, serveur radius, communication avec l'agent. Les tests seront faits le lundi 18 juillet.

**16h30 – 17h00** Rédaction du journal de travail et rangements.

## Lundi 18 juillet 2011

**09h00 – 12h00** Configuration et tests de l'architecture ASA/AD.

**13h00 - 16h30** Configuration et tests de l'architecture ASA/AD.

**16h30 – 17h00** Rédaction du journal de travail et rangements.

## Mardi 19 juillet 2011

**09h00 – 12h00** Rédaction du rapport.

**13h00 - 16h30** Rédaction du rapport.

**16h30 – 17h00** Rédaction du journal de travail.

## Mercredi 20 juillet 2011

**09h00 – 12h00** Rédaction du rapport. Partie annexe « Mise en place d'une architecture ASA/AD ».

**13h00 - 16h30** Rédaction du rapport : Résumé et introduction.

**16h30 – 17h00** Rédaction du journal de travail.

## Jeudi 21 juillet 2011

**09h00 – 12h00** Rédaction du rapport.

**13h00 – 16h30** Rédaction du rapport. (Conclusion)

**16h30 – 17h00** Rédaction du journal de travail.

## Vendredi 22 juillet 2011

**09h00 – 12h00** Rédaction du rapport.

**13h00 – 16h30** Relecture du rapport.

**16h30 – 17h00** Rédaction du journal de travail.

**Dimanche 24 juillet**

**2011**

**09h00 – 12h00** Rédaction du rapport.

**13h00 – 16h30** Rédaction du rapport.

**Lundi 25 juillet 2011**

**09h00 – 12h00** Rédaction du rapport.

**13h00 – 15h30** Rangements de tous les équipements utilisés lors de ce travail.

**Mardi 26 juillet 2011**

**09h00-12h00** Relecture finale.

**13h00 – 14h00** Impression, gravure du DVD, rendu.