Ce rapport est une étude préliminaire sur les thématiques suivantes : IPv6, DNS64, NAT64 et le L2TP

Projet intégré

IPv6 & Cloud

Groupe 4 – Julien Borgognon

1 Table of Contents

1 Introduction

Dans le cadre du cours « Réseau IP », il nous a été demandé, par groupe de 4 personnes, de réalisé un projet intégré. Nous sommes des ingénieurs du fournisseur d'accès *T2 Telecom.* Ce projet consiste à relier deux sites distants. Le premier site se trouvant à Fribourg sera équipé avec des appareils IP de version 4 ET 6. Tandis que le second site, situé à Berlin, sera entièrement en IPv6.

Les deux sites distants seront reliés avec un tunnel VPN. De plus, nous devons également mettre en place un cloud de type OpenStack dans les datacenters du fournisseur d'accès.

Le cahier de charge comporte les points suivants :

1. Utilisation native de IPv4 et IPv6
2. Attribution dynamique des adresses IPv4 et IPv6
3. Connexion à Internet à haut débit avec les deux protocoles cités en 1) pour le client (Attention) Berlin n'est qu'en IPv6 et doit pouvoir se connecter sur des sites IPv4 → NAT64/DNS64)
4. Routage dynamique entre les sites de Fribourg et Berlin (fournisseur d'accès)
5. Routage dynamique « privé » entre les sites de Fribourg et Berlin (client)
6. Mise en place d'un cloud de type OpenStack dans les datacenters du fournisseur d'accès, réparti entre le site de Fribourg et celui de Berlin, avec liaison de couche 2 entre les datacenters au travers du nuage L3.
7. Gestion du domaine DNS www.fri-learning.ch sur une machine virtuelle dans le cloud
8. Mise en place d'un serveur web www.fri-learning.ch comme vitrine de l'entreprise sur une machine virtuelle dans le cloud (une page d'accueil statique pour démontrer qu'il est accessible est suffisante)
9. Spécification de la structure du LAN et de l'accès sans fil au sein des bâtiments du siège principal du client
10. Spécification de l'architecture réseau de l'ISP sur son site de Berlin & Fribourg (datacenter,sécurité et connexion à Internet)

Dans cette étude préliminaire, les thématiques suivantes y sont décrises :

- IPv6
- DNS64
- NAT64
- VPN

Ce document comprend des descriptions et des synthèses sur les quatre fonctionnalités citées ci-dessus.

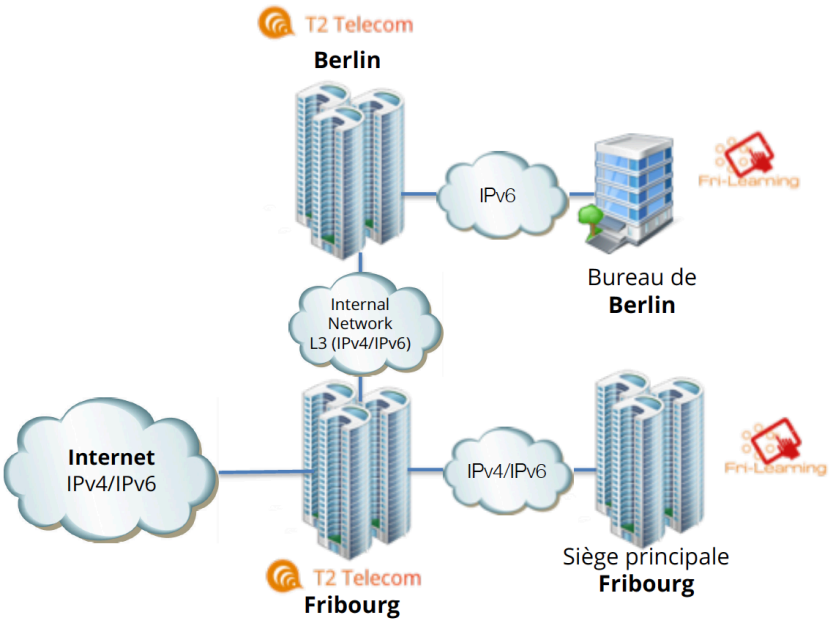

Figure 1 Réseau global de l'entreprise

2 Ipv6

IPv6 est l'acronyme d'Internet Protocole Version 6 qui est un protocole réseau de couche 3. IPv6 est l'évolution de l'IPv4. IETF a créé ce protocole afin de combler le manque d'adresses IPv4. En effet, les 4.3 milliards d'adresses IPv4 ont toutes été utilisées. La version 6 permet d'avoir 2^{128} adresses IP différentes (soit environ 3.4 x 10^{38}). Ce nombre abondant d'adresse permettra abandonner la traduction d'adresse dès que tout le monde sera passé en IPv6. En effet, tout le monde n'est pas encore sur la nouvelle version d'IP. C'est pourquoi nous devons utilisés des protocoles comme DNS64 et NAT64 qui seront expliqués plus tard dans ce rapport.

2.1 Structure

Les adresses IPv6 disposes de 128 bits contrairement à l'IPv4 qui en disposait « seulement » de 32. Ces 128 bits permettent d'avoir une énorme quantité d'adresses. Grâce à cela, la traduction d'adresse via le NAT n'est plus nécessaire.

Une adresse IPv6 dispose de 8 groupes de 2 octets qui sont séparés par le signe « deux points ». Les caractères composant l'adresse sont en hexadécimaux. Voici un exemple d'adresse IPv6 : *2001:0db8:0000:85a3:0000:0000:ac1f:8001.*

Cette adresse peut néanmoins être simplifiée de plusieurs manières afin de faciliter sa lecture et son écriture.

- Un bloc de 4 zéros consécutifs peut être simplifié en écrivant qu'un seul zéro. Cette pratique se fait aussi avec la version 4. L'adresse ci-dessus deviendra donc : *2001:0db8:0:85a3:0:0:ac1f:8001*
- Plusieurs blocs de zéros consécutifs peuvent être omis. Cependant il faut garder le signe « : » afin d'indiqué la simplification. Dans notre exemple, l'adresse deviendrait : *2001:db8:0:85a3::ac1f:8001*

Le préfixe indiquant le sous-réseau doit être compris entre 0 et 128. Le fonctionnement du calcul des adresses des sous-réseau reste le même qu'en IPv4.
Par exemple, le réseau *2001:db8:1f89::/48* contient les adresses entre *2001:db8:1f89:0:0:0:0:0 et 2001:db8:1f89:ffff:ffff:ffff:ffff:ffff*

Un deuxième exemple plus complexe :
Avec le réseau 2000::/3, nous avons comme adresses disponibles les adresses allant de *2000:0:0:0:0:0:0:0* à *3fff:ffff:ffff:ffff:ffff:ffff:ffff:ffff*

2.2 Explications

2000 :0 :0 :0 :0 :0 :0	Adresse du réseau
0010 0000 0000 0000:0 :0 :0 :0 :0 :0	Adresse représenté en binaire
1110 0000 0000 0000:0 :0 :0 :0 :0 :0	Masque de sous-réseau (/3)

Pour avoir la dernière adresse du sous-réseau, on inverse les bits du masque de sous-réseau et on les additionne à l'adresse du réseau de base :

0010 0000 0000 0000 :0 :0 :0 :0 :0 :0 :0
\+ 0001 1111 1111 1111 :1 :1 :1 :1 :1 :1 :1
0011 1111 1111 1111 :1 :1 :1 :1 :1 :1 :1. → 3fff: ffff :ffff :ffff :ffff :ffff :ffff :ffff

2.3 Quelques adresses particulières

Adresse non-spécifiée : ::/128 (tous les bits à 0) – Correspond à l'adresse 0.0.0.0 en IPv4. Cette adresse ne doit jamais être assigné à une station. Elle est utilisée comme adresse source par un nœud lors de son initialisation avant d'obtenir une adresse IP.

Localhost (loopback addresse): ::1/128 – Correspond à l'adresse 127.0.0.1 en IPv4.

2.4 En-tête

L'en-tête a une taille fixe de 40 octets et est largement simplifiée par rapport à celle de l'IPv4.

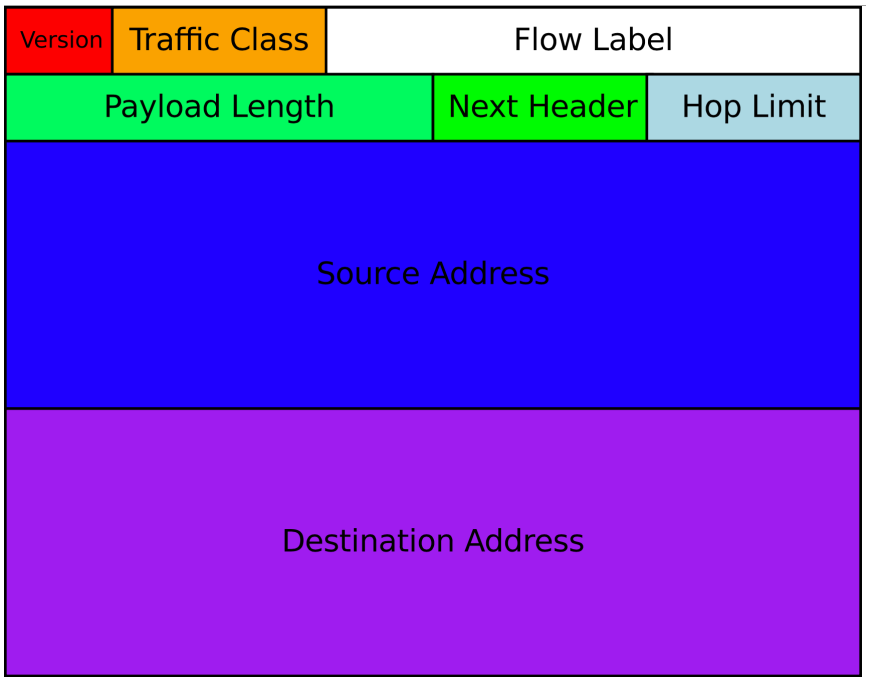

Figure 2 En-tête IPv6

Version : 4 bits, détermine le numéro du protocole internet

Traffic class : 8 bits, utilisé pour la QoS

Flow Label : 20 bits, permet de marquer le flux afin de le différencier

Payload length : 16 bits, taille de la charge utile en octets

Next header : 8 bits, indique le type de header qui suit immédiatement l'en-tête IPv6 (*cf en-tête d'extension*)

Hop limit : 8 bits, décrémenter de 1 à chaque passage dans un routeur. Le paquet est détruit lorsqu'il atteint 0. Correspond au TTL.

Source address : 128 bits, adresse source du paquet

Destination address : 128 bits, adresse destination du paquet

2.5 Principales différences avec l'en-tête IPv4

1. La taille de l'en-tête est fixe en IPv6 contrairement à l'IPv4. Le champs IP Header Length est donc inutile.
2. Le champs TTL est renommé en Hop Limit. Le champs TTL utilisait l'unité de temps [s] alors que le Hop Limit indique le nombre de routeur possible de « traverser ».
3. Les informations sur la fragmentation sont dans une autre en-tête

2.6 En-tête d'extension

Les en-têtes d'extension sont des options distinctes placées dans un paquet entre l'en-tête IPv6 et l'en-tête de la couche de transport comme on peut le voir sur l'image ci-dessous.

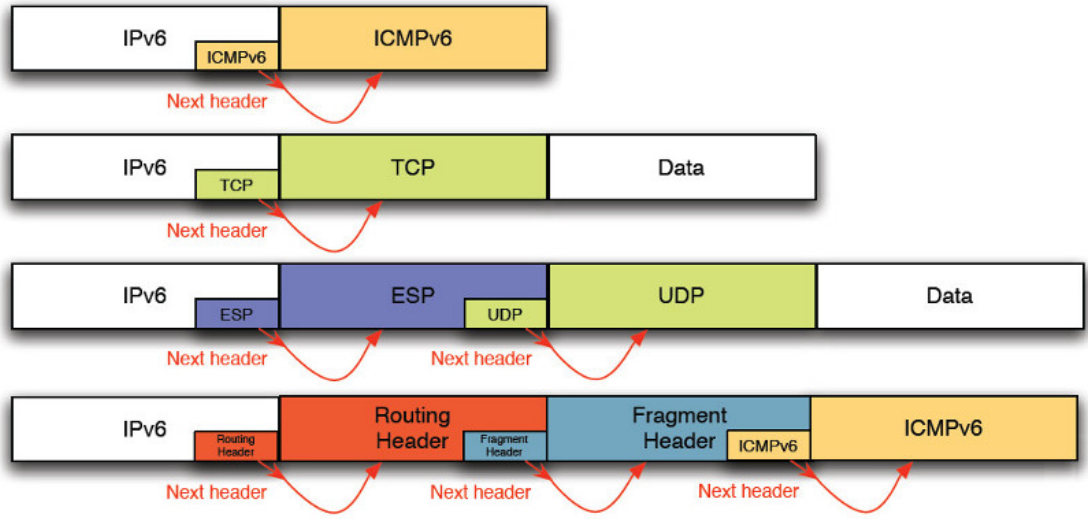

Figure 3 En-tête d'extension

La plupart des en-têtes d'extension ne sont pas traitées avant d'arriver à destination. Ce qui améliore grandement les performances des réseaux. Les routeurs n'ont pas besoin de vérifier toutes les options contrairement à l'IPv4.

Ces en-têtes ont une longueur indéfinie. Voici quelques types d'extension :

- Routage : permet la modification du routage à partir de la source
- Fragment : contient les informations concernant la fragmentation
- Authentication header : contient les informations pour l'authentification à IPSec et assure l'intégrité des données transférées
- Encapsulating Security Payload : permet de combiner plusieurs services de sécurité (IPSec)
- No Next Header : indique qu'il n'y a pas d'option qui suit

2.7 La portée de l'adresse

Il existe plusieurs types de communication en IPv6, nous allons les décrire dans ce sous-chapitre. Ces types décrivent l'étendue (scope) qui définissent dans quelle partie du réseau l'adresse peut communiquer et comment.

2.7.1 Unicast

L'unicast est le fait de communiquer à une seule et unique personne en même temps. Pour faire une analogie avec la vie réelle, c'est le fait d'envoyer une lettre postale à une seule personne.

La portée :

- Lien local : lorsque deux stations sont directement reliées
- Globale : des stations se situant dans le même réseau

2.7.2 Anycast

Une adresse anycast est un paquet qui est acheminé vers la station la plus rapide d'accès.

2.7.3 Multicast

Une adresse multicast est utilisée pour envoyer un seul paquet à plusieurs stations choisies.

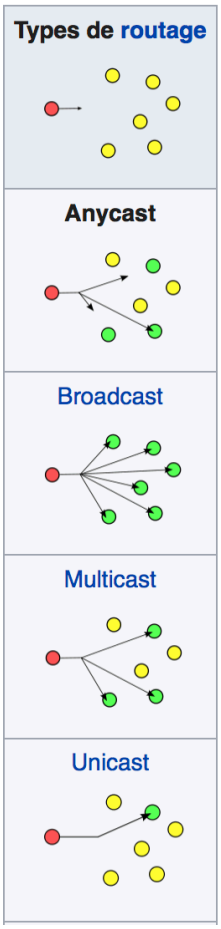

Figure 4
Types de routage

3 Transition d'IPv4 vers IPv6

Il y a trois options disponibles pour la migration au protocole IP version 6 :

- Dual-Stack
- Tunneling
- Traduction

3.1 Dual-Stack

Le Dual-Stack est une technologie de transition dans laquelle la version 4 et la version 6 d'IP cohabitent ensemble. Dans ce type de réseau, les deux versions d'IP sont déployées sur chaque équipement. Les protocoles déployés doivent également pouvoir gérer les deux versions.

Bien que cela parait une solution idéale, cette technologie contient un inconvénient. Il faut que le réseau soit capable de déployer l'IPv6 dans toute l'infrastructure. Pour cela, il est important de faire des mises à jour logicielles et matérielles importantes.

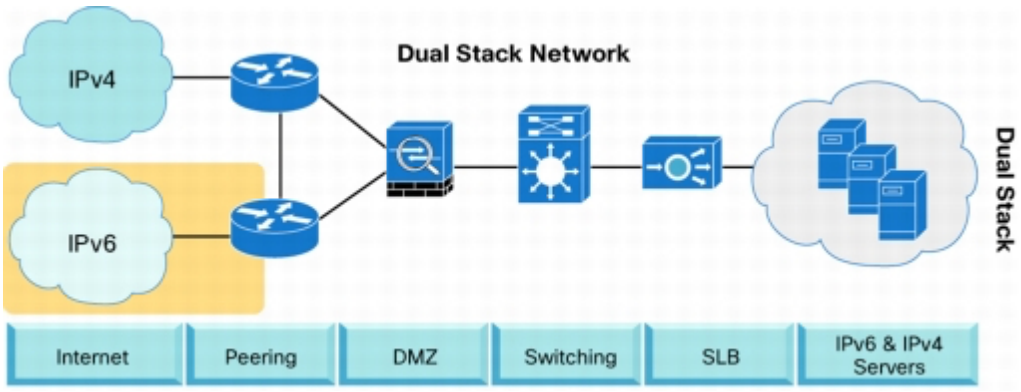

Figure 5 Réseau en Dual Stack

3.2 Tunneling

En utilisant l'option de tunneling, les paquets IPv6 traversent des segments reposant sur une topologie en version 4. L'avantage de cette approche est que les nouveaux protocoles peuvent fonctionner sans déranger les anciens protocoles. Cependant, il y a deux inconvénients conséquents :

- Les utilisateurs de la nouvelle infrastructure ne peuvent pas utiliser les services de l'architecture existante.
- Le tunnel n'autorise pas aux utilisateurs de la nouvelle infrastructure de communiquer avec les utilisateurs de l'ancienne s'ils n'ont pas de dual-stack.

Cette option comporte des inconvénients trop importants, elle ne sera pas plus approfondie

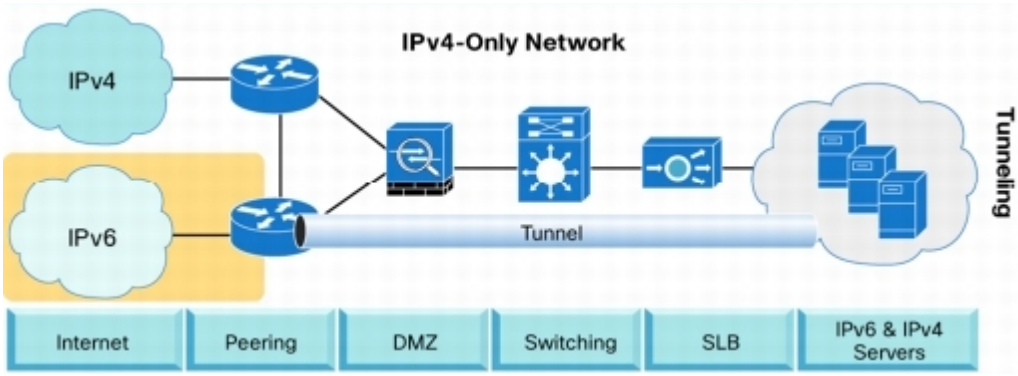

Figure 6 Réseau en Ipv4 avec un tunnel

.

3.3 Traduction

La traduction d'adresse facilite grandement la communication entre les deux versions IP. Cette option permet aux deux versions IP de communiquer entre elles grâce aux protocoles DNS64 et NAT64 qui seront décrit plus tard dans ce document. Cela parait la solution la plus optimale dans notre projet.

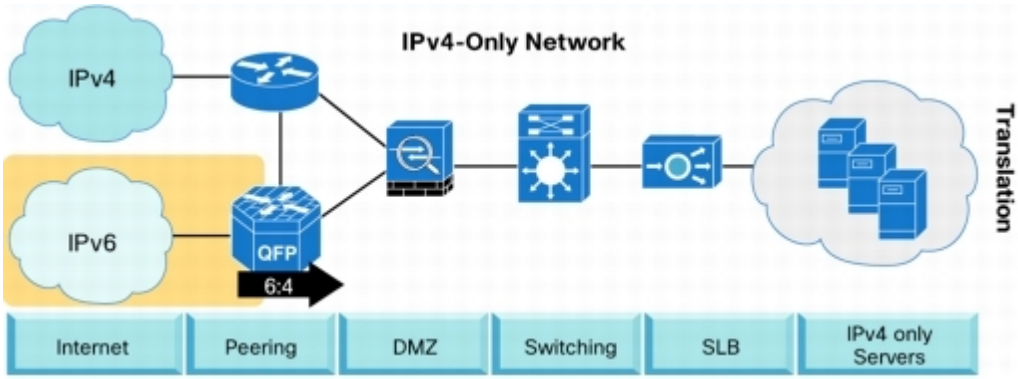

Figure 7 Réseau avec des appareils de traductions d'adresses

4 DNS64

4.1 Contexte

DNS64 est une nouvelle version du DNS standard. Le service DNS permet de traduire une adresse IP en un nom de domaine, et vice-versa. DNS est fondamentale pour une utilisation agréable d'internet car il est plus facile de retenir un nom qu'une suite de numéro. De plus, il est fréquent que les adresses IP de serveurs changent. Les noms de domaine ne sont, eux, pas modifiés. Cependant, il serait possible de n'utiliser que des adresses IP pour naviguer.

Le DNS64 a été introduit en même temps que l'IPv6. En effet, les machines purement IPv6 ne peuvent pas initier de communications avec les stations purement IPv4. Il faut donc fabriquer une adresse de type version 6 avec une adresse version 4 afin de pouvoir communiquer avec toutes les machines à travers d'un traducteur d'adresse (NAT64).

4.2 Fonctionnement

Supposons qu'un client possédant une seule adresse IPv6 veuille se connecter sur le site www.example.com avec comme adresse IP une adresse de version 4 : 203.0.113.1.

Voici un diagramme de séquence qui illustre les requêtes exécutées afin d'obtenir l'adresse IPv6 de destination :

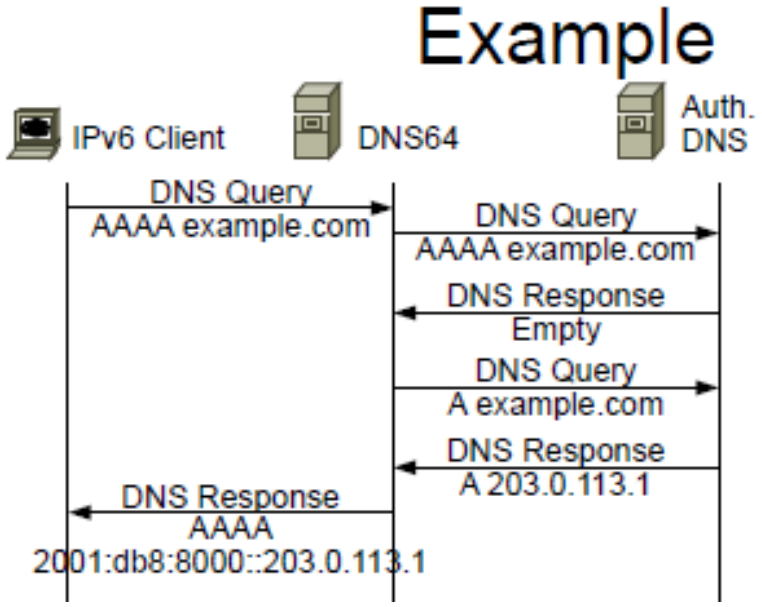

Figure 8 Diagramme de séquence d'une requête DNS64

1. Le client envoie une requête de type AAAA à son serveur DNS.
2. Le serveur DNS64 contacte le serveur DNS autoritaire afin d'obtenir une adresse IPv6 correspondant à www.example.com
3. Le serveur autoritaire ne possède pas cette adresse. Il envoie donc une réponse vide. En réalité, le serveur a une correspondance pour le nom de domaine mais n'a pas d'enregistrement AAAA en rapport
4. Le serveur DNS64 envoie alors une requête de type A au serveur autoritaire
5. Le serveur autoritaire trouve une correspondance dans sa table d'adresses et lui répond avec l'adresse IPv4 du serveur WEB : 203.0.113.1
6. Le serveur DNS64 converti l'adresse IPv4 en IPv6 de manière algorithmique. Ensuite il peut répondre au client l'adresse IPv6

Un préfixe est réservé par le DNS64 et le NAT64 pour la conversion des adresse IPv4 en IPv6. Ce préfixe ne doit pas excéder 96 bits ! Si aucun préfixe n'est configuré sur ces serveurs, le préfixe par défaut est : 64:ff9b::/96.

Ensuite, les 32 bits restant (128 – 98 = 32 bits) de l'adresse sont complétés par l'adresse IPv4. Dans notre exemple, le préfixe configuré est 2001:db8:8000::/96.

Cela nous fait comme adresse finale : 2001:db8:8000::203:0:113:1

5 NAT64

Le protocole NAT fait correspondre des adresses IP à d'autres adresses IP. Ce mécanisme est utilisé afin de permettre à des stations ayant des adresses IP privées d'avoir une adresse IP publique pour communiquer sur Internet. Il est possible de faire correspondre plusieurs adresses IP privée à une seule adresse IP publique. Ceci permet d'atténuer l'épuisement des adresses IPv4.

Le NAT64 travaille différemment.

5.1 Contexte

Le NAT64 est un mécanisme qui facilite la traduction entre les adresses IPv4 et IPv6. Cependant il n'est pas suffisant pour permettre à deux protocoles IP différents de communiquer. Il doit être mis en parallèle avec le DNS64.

5.2 Structure

Une installation basique du NAT64 fait office de passerelle entre un réseau IP version 4 et un réseau IP version 6. Le trafic provenant du réseau IP version 6 est routé par la passerelle qui gère toutes les traductions d'adresses pour que le transfert de paquets se fasse, et vice-versa.

Il y a deux types de NAT64, le Stateless NAT64 et le Stateful NAT64.

5.2.1 Stateless NAT64

Ce mode permet de traduire une adresse IPv4 en une adresse IPv6, et vice-versa. La traduction implique l'analyse de l'en-tête IPv6 en incluant les en-têtes d'extension, extrait les informations pertinentes et les convertis en une en-tête IPv4. Ce processus se fait aussi pour passer de l'IPv4 à l'IPv6. Avec ce mode, il n'y a aucune économie d'adresse, juste une association.

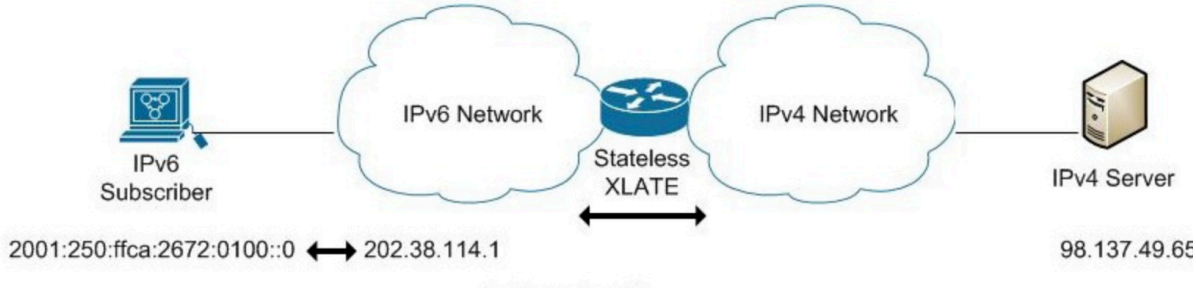

Figure 9 Stateless NAT64

5.2.2 Stateful NAT64

Ce mode agit comme le PAT, il permet d'associer une ou plusieurs adresses IPv6 à une seule adresse IPv4. En plus de cela, un port est associé à l'adresse. L'adresse IPv4 reste la même, mais le port source change à chaque nouvelle station. C'est ce port qui permet de retrouver la machine qui a envoyé les données.

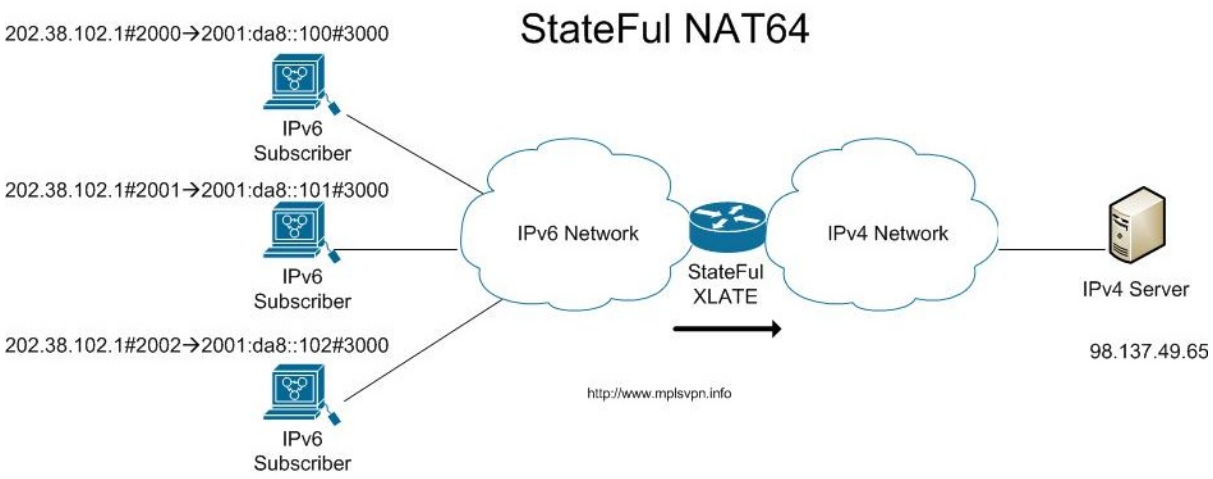

Figure 10 Stateful NAT64

6 DNS64 et NAT64 dans un réseau

La communication entre deux réseaux IP de version différente se fait à travers la combinaison des protocoles NAT64 et DNS64.

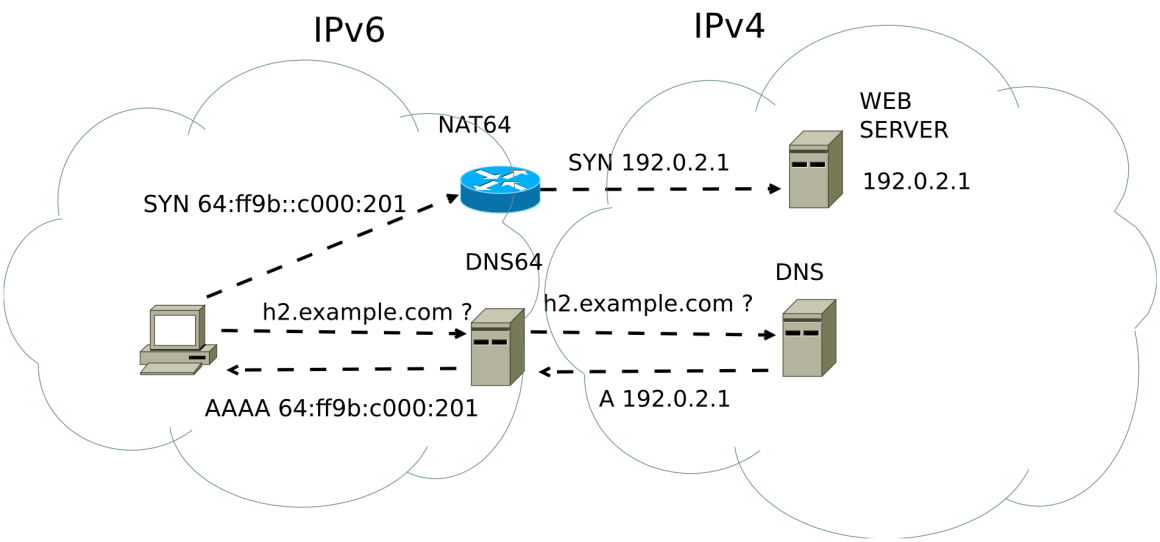

Figure 11 Cohabitation du NAT64 et du DNS64 dans un réseau

Le client en IPv6 veut initier une session sur le serveur WEB en IPv6.

1. Le client fait une requête DNS afin de récupérer l'adresse IP du serveur (cf DNS64). L'adresse reçu est une adresse de version 6.
2. La station initie la session en direction du serveur
3. Le routeur NAT64 associera l'adresse IPv6 reçue à une adresse de son pool d'adresse IPv4 s'il est en Stateless. S'il est en Stateful, il enregistrera le port associé à l'adresse IPv6 à l'adresse IPv4 publique qu'il possède.
4. Le serveur reçoit le paquet. S'il répond, sa réponse sera également traduite par le NAT64 en IPv6.

Il est important de noter que ces traductions se font de manière totalement transparente. Les stations ne perçoivent pas de changements.

7 VPN

Les VPN ou Virtual Private Network permettent de créer un tunnel virtuel entre deux réseaux physiques distants. Ce tunnel est totalement transparent pour l'utilisateur. Il existe plusieurs protocoles offrant ce service. Ils se diffèrent par leur rapidité, leur utilisation, leur sécurité ou leur facilité d'utilisation. Dans notre réseau, un tunnel VPN sera créé entre Fribourg et Berlin afin que les deux sites puissent communiquer.

Je vais présenter les principaux services VPN et approfondir le meilleur.

7.1 PPTP

Le Point-to-Point Tunneling Protocol ou PPTP est basé sur le PPP et est souvent utilisé de part sa facilité d'utilisation. C'est un protocole de couche 2 et utilise le port TCP 1723. Il utilise un tunnel GRE pour encapsuler les paquets PPP.

LE PPTP crée un tunnel privé pour envoyer des données depuis un appareil distant. L'authentification se fait uniquement par un mot de passe. Les données transitées du tunnel ne sont pas cryptées.

Avantages	Inconvénients
Multiplateforme	Mal sécurisé
Simple à utiliser et à mettre en place	
Système rapide	

7.2 Open VPN

Open VPN est un application open source avec ces connexions PPP sécurisées. Il utilise le SSL/TLS. Il permet aux clients de se connecter avec des clés pré-partagées, des certificats ou avec un nom d'utilisateur et un mot de passe. Ce service fonctionne sous UDP ou TCP. En UDP, il fait preuve d'une grande stabilité et d'une vitesse optimale.

Avantages	Inconvénients
Système rapide	Difficile d'installation (client ET serveur)
Sécurité robuste	
Mise à jour fréquente	

7.3 GRE

Le GRE est un protocole d'encapsulation qui encapsule plusieurs protocoles dans des paquets IP. Il se situe à la couche 3 du modèle OSI. Le *Generic Routing Encapsulation* crée une liaison virtuelle point à point au-dessus d'un réseau.

Une en-tête GRE supplémentaire est ajouté aux paquets transportés. Cette en-tête contient différents flags et le type de protocole qui est transporté. Ce protocole n'est pas du tout sécurisé, il n'offre pas d'authentification des données, ni de contrôle d'intégrité.

Figure 12 Exemple d'un tunnel GRE

7.4 L2TP

Ce protocole d'encapsulation permet d'encapsuler des paquets PPP sur les couches 2 ou 3 du modèle OSI. Ce protocole est basé sur le PPTP et sur le L2F. Il n'en a gardé que leurs avantages (rapidité et fiabilité). En revanche, son désavantage est la taille des en-têtes des paquets encapsulés. En effet, cette accumulation de protocoles encapsulés (IP/PPP/L2TP/UDP) peut rajouter jusqu'à 50 octets d'informations supplémentaire.

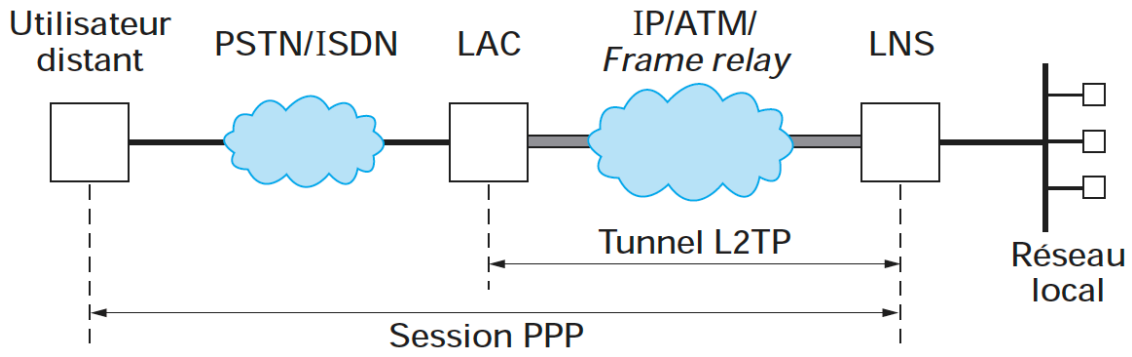

Figure 13 Schéma basique d'une infrastructure L2TP

7.4.1 Terminologie

7.4.1.1 L2TP Access Concentrator : LAC

Le LAC est l'équipement terminal du tunnel L2TP qui interagit pour l'utilisateur. C'est lui qui reçoit les paquets PPP de l'utilisateur pour ensuite les encapsuler et les envoyer à l'autre extrémité du tunnel, vers le LNS. Il fait l'opération inverse lorsqu'il reçoit des paquets du LNS. Il retire l'encapsulation avant de les transmettre à la station distante

7.4.1.2 L2TP Network Server : LNS

Le serveur LNS est la deuxième extrémité du tunnel. Il communique directement avec le LAC via le tunnel. Le LNS accomplit la pratiquement la même fonction que le LAC. Il ajoute, respectivement retire, l'encapsulation du L2TP des paquets qui transitent. La différence avec le LAC est qu'il est le point de terminaison logique de la session PPP. Le LNS est également responsable de l'authentification du tunnel.

7.4.2 Tunnel L2TP

Le tunnel L2TP se fait entre deux nœuds LAC et LNS. Il contient une connexion de contrôle et zéros ou plusieurs sessions. C'est grâce à ce tunnel que les paquets sont échangés et que les paquets de contrôle sont échangé.

7.4.3 Session L2TP

Les sessions L2TP sont créées à l'intérieur du tunnel. Ce sont les stations distantes qui les initient en communiquant avec le LNS. Il peut en exister autant qu'il y a de stations communiquant avec le LNS.

7.4.4 Connexion de contrôle

La connexion de contrôle gère la communication du tunnel et est établie entre le LAC et le LNS avant même que le tunnel ne soit créé. Elle gère la construction, respectivement la destruction du tunnel et des sessions.

7.4.5 Description du protocole

Afin qu'une session PPP puisse être transportée jusqu'à un LNS, le LAC doit demander la création d'un tunnel. Ensuite, le LNS envoie une confirmation de la création du tunnel. Pour finir le LAC répond également avec une confirmation du tunnel. Cette méthode est semblable au fameux « Three-Way Handshake » du TCP.

Une fois la session PPP créée, on établit la session L2TP en envoyant des messages de contrôle de façon similaire (demande → première confirmation → seconde confirmation). Dès lors, la connexion L2TP est établie. Le protocole envoie des messages de contrôle régulièrement dans le but de s'assurer de la bonne communication à travers le tunnel. Ces messages doivent être acquitté.

Dans ce tunnel, il y a deux types de messages qui transitent : les messages de données et les messages de contrôle. Les messages de données contiennent les trames PPP encapsulées qu'enverra le client. Ils sont transportés sur un canal peu fiable en UDP.
Les messages de contrôles gèrent l'établissement, la continuité et la destruction du tunnel et des sessions. Ils sont transportés à l'aide d'un canal fiable gérant la retransmission.

7.4.6 En-tête L2TP

Les messages de contrôle et de données ont la même en-tête. Seules les options varient.

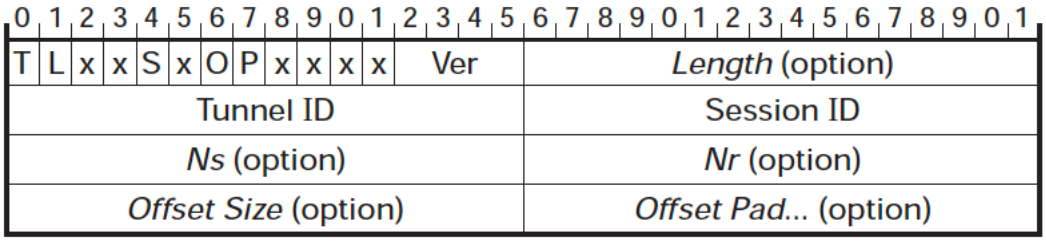

Figure 14 En-tête L2TP

T : indique le type de message. '0' pour un message de données, '1' pour un message de contrôle

L : indique si le champ Lenght est présent par la valeur '1'

x : bits réservés pour des extensions futures. Ils sont donc par défaut à '0'

S : indique si les champs Ns et Nr sont présents

O : indique si le champ Offset Size est présent

P : indique avec la valeur '1' si le message de données doit être traité de façon privilégiée

Ver : indique la version du protocole utilisé

Length : indique la longueur totale du message en octet

Tunnel ID : indique l'identifiant du tunnel pour la connexion de contrôle. Il est possible qu'un même tunnel ait deux ID car le LAC et le LNS attribuent chacun un ID.

Session ID : indique l'identifiant de session dans un tunnel. Il est possible qu'une même session ait deux ID car le LAC et le LNS attribuent chacun un ID.

Ns : indique le numéro de séquence pour un message de données ou de contrôle

Nr : indique le numéro de séquence attendu pour le prochain message de contrôle. Ce champ est ignoré s'il est présent dans un message de données

Offset Size : indique le nombre d'octets dans un Offset Padding. C'est-à-dire le nombre d'octets qu'il y a entre la fin de l'en-tête L2TP et le début des données PPP-

Offset Padding : ce champ n'est pas défini pour l'instant

8 Conclusion

L'élaboration de ce rapport m'a permis d'acquérir beaucoup d'informations et de savoir sur les thématiques abordées, à savoir : l'IPv6, le DNS64, le NAT64 et les VPN.

Je suis persuadé que ces informations seront utiles pour la suite du projet et pourront aidées à mener à bien notre projet.

Cette grande charge de travail condensée sur une courte période de temps m'a permis de m'améliorer au niveau de la planification.

Julien Borgognon

9 Glossaire

Anagramme	Nom complet
DNS	Domain Name System
DNS64	Domain Name System from IPv6 to IPv4
GRE	Generic Routing Encapsulation
IETF	Internet Engineering Task Force
IP	Internet Protocole
IPSec	Internet Protocole Security
IPv4	Internet Protocole version 4
IPV6	Internet Protocole version 6
L2F	Layer 2 Forwarding Protocole
L2TP	Layer 2 Tunneling Protocol
LAC	L2TP Access Concentrator
LNS	L2TP Network Server
NAT64	Network Address Translation from IPv6 to IPv4
OSI	Open System Interconnexions
PAT	Port Address Translation
PPP	Point-to-Point Protocole
PPTP	Point-to-Point Tunneling Protocole
QoS	Quality of Service
Requête de type A	Requête DNS pour les adresses IPv4
Requête de type AAAA	Requête DNS pour les adresses IPv6
SSL	Secure Sockets Layer
TCP	Transmission Control Protocol
TLS	Transport Layer Security
TTL	Time To Live
Tunneling	Encapsulation
UDP	User Datagram Protocol
VPN	Virtual Private Network

10 Références

10.1 IPV6 :

- ✓ https://fr.wikipedia.org/wiki/IPv6
- ✓ http://www.ipuptime.net/Unspecified.aspx
- ✓ https://docs.oracle.com/cd/E19957-01/820-2982/ipv6-ref-3/index.html
- ✓ https://en.wikipedia.org/wiki/IPv6_address#Address_scopes
- ✓ https://reussirsonccna.fr/unicast-multicast-broadcast-oui-mais-quelle-couche/
- ✓ Image : http://livre.g6.asso.fr/images/f/ff/2015_10_10_extensions_IPv6_v01.jpg

10.2 VPN :

- ✓ https://www.astrill.com/fr/vpn-protocols
- ✓ http://www.supinfo.com/articles/single/557
- ✓ https://www.le-vpn.com/fr/protocoles-le-vpn/

10.2.1 L2TP :

- ✓ Protocole L2TP : par Étienne GALLET DE SANTERRE
- ✓ http://www.frameip.com/l2tp-pppoe-ppp-ethernet/

10.3 DNS64 :

- ✓ https://tools.ietf.org/html/rfc6147
- ✓ http://www.bortzmeyer.org/6147.html

10.4 NAT64 :

- ✓ https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/ipaddr_nat/configuration/xe-3s/nat-xe-3s-book/iadnat-stateless-nat64.pdf

10.5 Transition IPv4 à IPv6 :

- ✓ https://www.cisco.com/c/en/us/products/collateral/ios-nx-os-software/enterprise-ipv6-solution/white_paper_c11-676278.html

11 Table de figures