Réseaux IP

Etudes Préliminaires

Auteurs :
M. Maic Queiroz

Encadrant :
M. François Buntschu

24 mars 2017

Table des matières

1 Introduction

Ce rapport a pour objectif de décrire les différents thèmes et technologies nécessaires à la réalisation de ce projet. Les thèmes suivants seront traités dans ce document :

- Adressage IP (v4 et v6)
- Liaison de couche 2 - L2TPv3
- Monitoring et analyse des flux (outil NetFlow)

Dans cette analyse préliminaire seront décrites de manière théorique les technologies mentionnées ci-dessus. Cette description pourra comprendre le fonctionnement, le concept ainsi que les spécifications de ces technologies.

Pour rappel, le projet consiste à répondre au mandat que la société **Fri-Thinking & Co** nous a confié. En plus de l'accès Internet haut débit, les services et tâches suivants sont demandés dans le cahier de charge du projet :

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 au point 1 - Bulle n'est qu'en IPv6 et doit pouvoir se connecter sur des sites IPv4 à NAT64/DNS64
4. Routage dynamique entre le site de Fribourg et Bulle (fournisseur d'accès)
5. Routage dynamique « privé » entre le site de Fribourg et Bulle (client)
6. Mise en place d'un cloud de type OpenStack réparti entre le site de Fribourg et celui de Bulle, avec liaison de couche 2 entre les datacenters au travers du nuage L3.
7. Gestion du domaine DNS fri-thinking.ch sur une machine virtuelle dans le cloud
8. Mise en place d'un serveur web www.fri-thinking.ch comme vitrine de l'entreprise sur une machine virtuelle dans le cloud
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 Bulle & Fribourg (datacenter, sécurité et connexion à Internet)
11. Supervision du trafic inter sites au moyen de la fonctionnalité « NetFlow ».

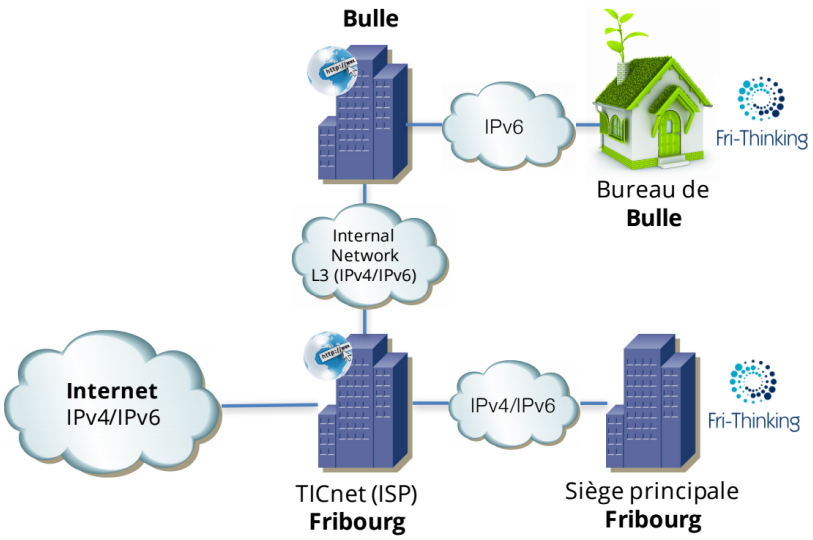


FIGURE 1 – Schéma du projet

2 Adressage IP

L'adressage est une fonction essentielle des protocoles de couche réseau (couche 3 du modèle OSI). Il permet la transmission de données entre des hôtes situés sur un même réseau ou sur des réseaux différents.
IP utilise des adresses numériques appelées adresses IP, composées respectivement de 32 bits (IPv4) et de 128 bits (IPv6).
L'ICANN (Internet Corporation for Assigned Names and Numbers) est chargée d'attribuer des adresses IP publiques (adresses IP des équipements directement connectés sur le réseau public Internet).

2.1 IPv4

2.1.1 Généralités

Dans le cas d'IPv4 les adresses sont composées de 32 bits, que l'on représente généralement sous la forme de quatre nombres entiers séparés par des points où chaque nombre représente un octet. Exemple : 192.168.1.10
La plage d'adressage est donc comprise de 0.0.0.0 à 255.255.255.255, cependant certaines adresses ne peuvent être utilisées par les hôtes (adresse de broadcast, localhost, adresse de réseau, etc.).

2.1.2 Décomposition d'une adresse IPv4

Une adresse IPv4 de 32 bits se compose de deux parties :
- Une partie réseau (net ID) composée d'un certain nombre de bits à gauche de l'adresse qui désignent le réseau.
- Une partie hôte (host ID) composée d'un certain nombre de bits à droite de l'adresse qui désignent les hôtes de ce réseau.

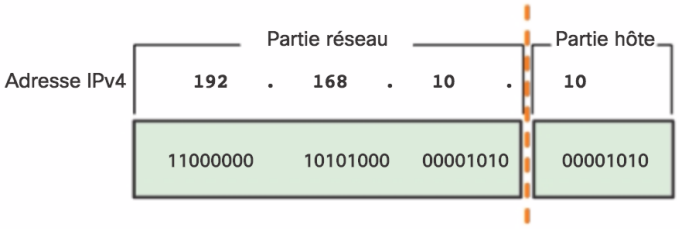


FIGURE 2 – Parties composant une adresse IPv4

La séparation entre la partie réseau et la partie hôte s'effectue à l'aide du masque réseau. Cependant, le masque réseau ne contient pas d'adresse réseau à proprement parler, il permet uniquement d'indiquer quels sont les bits représentant le réseau dans une adresse IPv4.

Ainsi, un bit valant '1' dans le masque réseau indique que, le bit se trouvant à la même position dans l'adresse de l'hôte, est un bit réseau. Inversement, un bit à '0' dans le masque réseau indique qu'il s'agit d'un bit d'hôte.
De cette manière, l'opération AND bit à bit entre le masque réseau et l'adresse IPv4 de l'hôte permet d'obtenir l'adresse réseau.

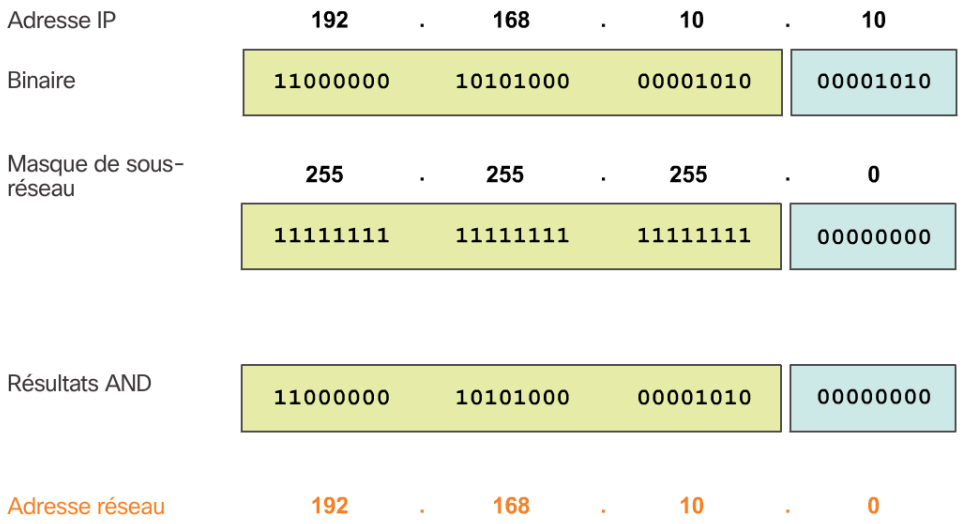


FIGURE 3 – Calcul de l'adresse de réseau (bitwise AND)

Remarque : L'adresse réseau (résultat de l'opération AND bit à bit entre le masque et l'adresse IP de l'hôte) est identique pour l'ensemble des hôtes se trouvant sur un même réseau.

2.1.3 Masque de sous-réseau - Préfixe

Exprimer dans un plan d'adressage les adresses d'hôtes avec l'adresse du masque de sous-réseau correspondant au format décimal à point (dotted decimal notation) peut devenir long et fastidieux. Il existe donc une méthode plus rapide appelée longueur de préfixe (prefix length).
La longueur de préfixe correspond au nombre de bits définis à '1' dans le masque de sous-réseau. Elle est notée au moyen de la notation « /n » où n correspond au nombre de bits à '1'.

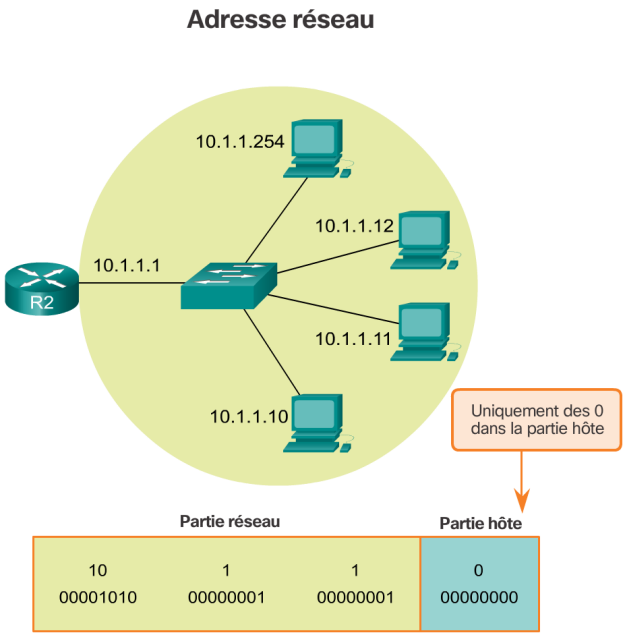
Voici quelques exemples :

Masque de sous-réseau	Adresse 32 bits	Longueur de préfixe
255.0.0.0	**11111111**.00000000.00000000.00000000	/8
255.255.0.0	**11111111.11111111**.00000000.00000000	/16
255.255.255.0	**11111111.11111111.11111111**.00000000	/24
255.255.255.128	**11111111.11111111.11111111.1**0000000	/25
255.255.255.192	**11111111.11111111.11111111.11**000000	/26
255.255.255.224	**11111111.11111111.11111111.111**00000	/27
255.255.255.240	**11111111.11111111.11111111.1111**0000	/28
255.255.255.248	**11111111.11111111.11111111.11111**000	/29
255.255.255.252	**11111111.11111111.11111111.111111**00	/30

FIGURE 4 – Quelques préfixes

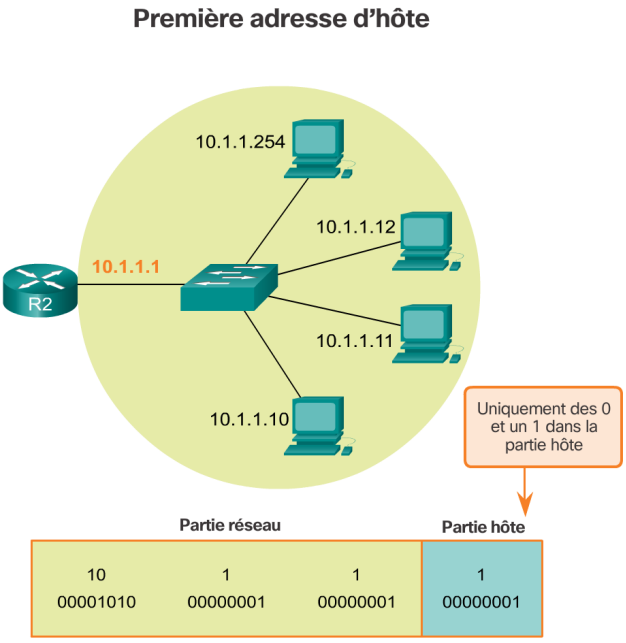
2.1.4 Adresses réseau, d'hôte et de diffusion

Si l'on connaît une adresse IP d'un réseau ainsi que son masque, on peut calculer une multitude d'adresses en remplaçant les bits de la partie hôte comme suit :



La figure 5 démontre qu'il est possible de calculer l'adresse de réseau en mettant l'ensemble des bits de la partie hôte à '0'. L'adresse réseau ainsi obtenue est 10.1.1.0 /24.

FIGURE 5 – Adresse du réseau



La figure 6 démontre qu'il est possible de calculer l'adresse du premier hôte en mettant l'ensemble des bits de la partie hôte à '0' sauf le dernier (LSB) qui lui sera à '1'. L'adresse du premier hôte ainsi obtenue est 10.1.1.1.

Remarque : Cette adresse est généralement attribuée à l'interface du routeur et devient donc la passerelle par défaut de tous les hôtes sur ce réseau. Ceci est une pratique courante, mais nullement une obligation.

FIGURE 6 – Adresse du premier hôte

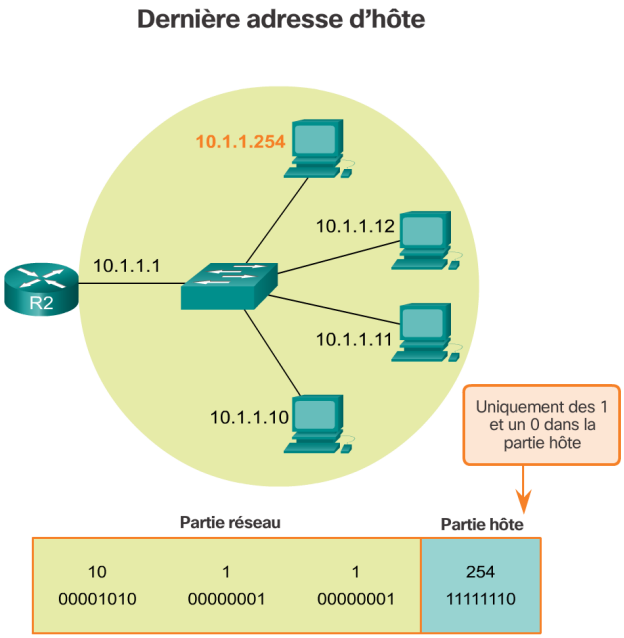


FIGURE 7 – Adresse du dernier hôte

La figure 7 démontre qu'il est possible de calculer l'adresse du dernier hôte en mettant l'ensemble des bits de la partie hôte à '1' sauf le dernier (LSB) qui lui sera à '0'. L'adresse du premier hôte ainsi obtenue est 10.1.1.254.

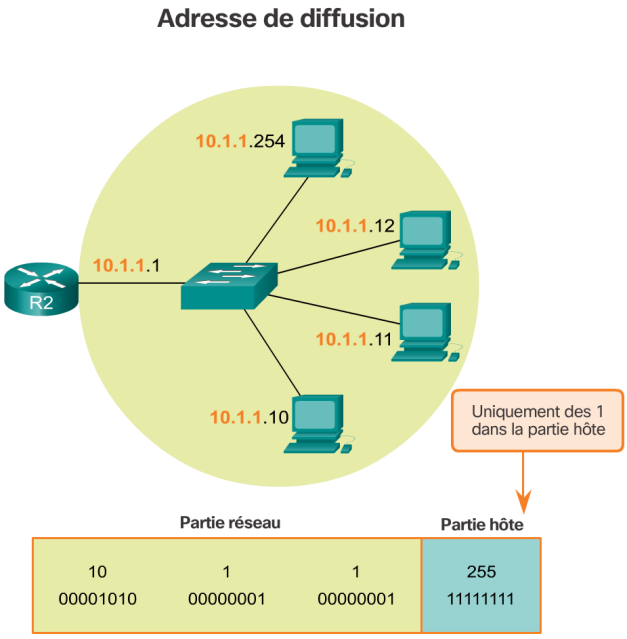


FIGURE 8 – Adresse de broadcast

La figure 8 démontre qu'il est possible de calculer l'adresse de diffusion (broadcast) de notre réseau en mettant l'ensemble des bits de la partie hôte à '1'. L'adresse réseau ainsi obtenue est 10.1.1.255.

Remarque : Une définition spéciale existe pour l'adresse de diffusion 255.255.255.255. Il s'agit de l'adresse de diffusion du réseau 0.0.0.0, qui, dans les normes de protocole Internet, représente le réseau local mais sur une plage d'adresse comprise de 0.0.0.0 à 0.255.255.255 (0.0.0.0 /8). La transmission à cette adresse est limitée par définition, en ce sens qu'elle n'est jamais transmise par les routeurs connectant le réseau local à d'autres réseaux.

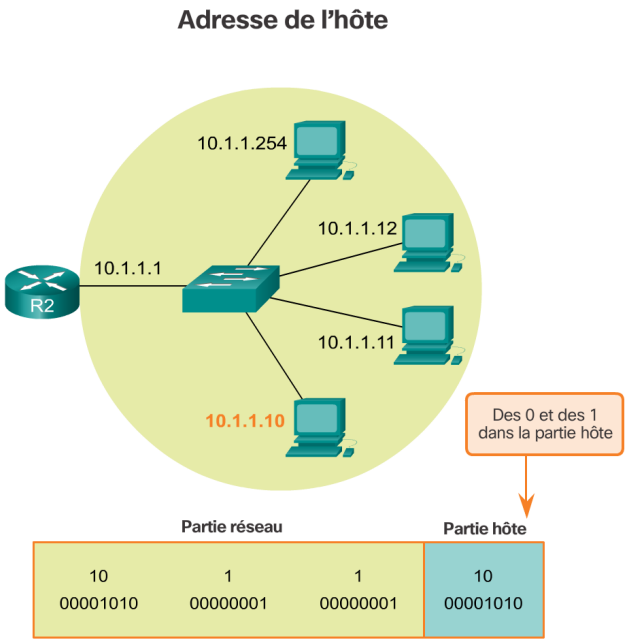


FIGURE 9 – Adresse d'hôtes

Pour finir, la figure 9 démontre que toutes les autres combinaisons de bits à '1' ou à '0' que celles susmentionnées donnent une adresse d'hôte.

2.1.5 Types de communication

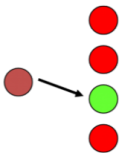


FIGURE 10 – Monodiffusion

La transmission monodiffusion (unicast) est utilisée dans les communications d'hôte à hôte et client-serveur. Les paquets unicast utilisent l'adresse de l'hôte de destination comme adresse de destination.
Le paquet est donc destiné à un seul hôte comme le montre la figure 10.

La transmission multidiffusion (multicast) permet de réduire le volume du trafic. En effet, elle permet à un hôte d'envoyer un seul paquet à un groupe de multidiffusion (groupe d'hôtes) comme le montre la figure 11.
La plage d'adresse 224.0.0.0 à 239.255.255.255 est réservée comme plage multicast dans IPv4.
Les adresses de la plage 224.0.0.0 à 224.0.0.255 sont réservées à la multidiffusion sur le réseau local et sont donc principalement utilisées par les protocoles de routage pour la transmission des informations de routage (224.0.0.9 pour le protocole RIP version 2).

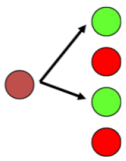


FIGURE 11 – Multidiffusion

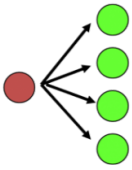


FIGURE 12 – Diffusion

La transmission de diffusion (broadcast) est utilisée pour envoyer des paquets à l'ensemble des hôtes du réseau à l'aide de l'adresse de broadcast comme le montre la figure 12.
L'adresse de destination du paquet a pour caractéristique d'avoir uniquement des '1' dans la partie hôte. Tous les hôtes se trouvant sur ce réseau local recevront le paquet et le regarderont.
Lorsqu'un hôte reçoit un paquet envoyé à l'adresse de diffusion du réseau, il traite le paquet comme s'il s'agissait d'un paquet adressé à son adresse de monodiffusion.

2.1.6 Adresses privées / publiques

Les adresses IPv4 publiques sont acheminées de manière globale sur les routeurs Internet. En raison de la pénurie d'adresses IPv4 publiques, des adresses IPv4 privées ont été créées. Ces adresses ont pour caractéristique principale de ne pas être uniques. Elles sont donc utilisées par un réseau interne pour l'adressage des hôtes.

Les blocs d'adresses privées sont les suivants :

- **10.0.0.0 /8** : 10.0.0.0 à 10.255.255.255
- **172.16.0.0 /12** : 172.16.0.0 à 172.31.255.255
- **192.168.0.0 /16** : 192.168.0.0 à 192.168.255.255

Les adresses appartenant à ces blocs ne sont pas autorisées sur Internet et doivent être filtrées (rejetées) par les routeurs Internet. Si l'on prend pour exemple la figure 13, on peut constater que les utilisateurs des réseaux 1, 2 ou 3 envoient des paquets à des destinations éloignées. Les routeurs du FAI voient que les adresses IPv4 sources des paquets sont des adresses privées et rejettent donc les paquets.

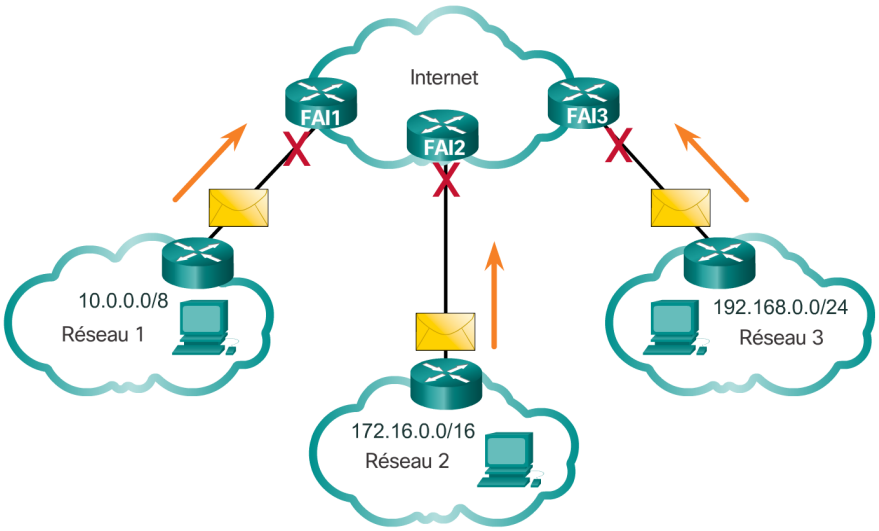


FIGURE 13 – Rejet des paquets dont l'adresse IPv4 source est privée

Les adresses IPv4 privées sont donc utilisées pour adresser des hôtes internes. Mais, ces adresses ne sont pas routables via Internet et doivent être traduites en adresses IPv4 publiques à l'aide de la traduction d'adresses réseau (NAT).

2.1.7 Adresses particulières

- **Adresses de bouclage (Loopback addresses)** :
  Plage comprise de 127.0.0.1 à 127.255.255.254 (127.0.0.0 /8). Ce sont des adresses spéciales utilisées par des hôtes pour diriger le trafic vers eux-mêmes. Par exemple, elles peuvent être utilisées sur un hôte pour vérifier si la configuration TCP/IP est opérationnelle
- **Adresses link-locales (Link-Local addresses)** :
  Plage comprise de 169.254.0.1 à 169.254.255.254 (169.254.0.0 /16). Ces adresses sont connues sous le nom d'adresses APIPA (adressage IP privé automatique), elles sont utilisées par un client DHCP pour se configurer automatiquement si aucun serveur DHCP n'est disponible.
- **Adresses TEST-NET (TEST-NET adresses)** :
  Plage comprise de 192.0.2.0 à 192.0.2.255 (192.0.2.0/24), Ces adresses sont réservées à des fins pédagogiques (utilisées dans la documentation et dans des exemples de réseau).

— **Adresses expérimentales (Experimental Addresses)** :
Plage comprise de 240.0.0.0 à 255.255.255 .254 réservées pour une utilisation future

2.1.8 Adressage avec et sans classe

Historiquement les adresses IPv4 Internet étaient attribuées à l'aide de l'adressage par classe (RFC 790). Dans la RFC, les plages de monodiffusion sont divisées en classes spécifiques :

— **Classe A** (0.0.0.0/8 à 127.0.0.0/8) : Créée pour prendre en charge les réseaux de très grande taille.
— **Classe B** (128.0.0.0 /16 à 191.255.0.0 /16) : Créée pour répondre aux besoins des réseaux de taille moyenne.
— **Classe C** (192.0.0.0 /24 à 223.255.255.0 /24) : Créée pour répondre aux besoins des réseaux de petite taille

Spécifications de la classe A	
Bloc d'adresses	0.0.0.0 à 127.0.0.0*
Masque de sous-réseau par défaut	/8 (255.0.0.0)
Nombre maximal de réseaux	128
Nombre d'hôtes par réseau	16 777 214
Bit d'ordre haut	0xxxxxxx.___.___.___

* 0.0.0.0 et 127.0.0.0 sont réservées et ne peuvent pas être attribuées

Spécifications de la classe B	
Bloc d'adresses	128.0.0.0 à 191.255.0.0
Masque de sous-réseau par défaut	/16 (255.255.0.0)
Nombre maximal de réseaux	16 384
Nombre d'hôtes par réseau	65 534
Bit d'ordre haut	10xxxxxx.___.___.___

Spécifications de la classe C	
Bloc d'adresses	192.0.0.0 à 223.255.255.0
Masque de sous-réseau par défaut	/24 (255.255.255.0)
Nombre maximal de réseaux	2 097 152
Nombre d'hôtes par réseau	254
Bit d'ordre haut	110xxxxx.___.___.___

FIGURE 14 – Classes principales et leurs spécificités

Remarque : il existe également un bloc d'adresses de multidiffusion (classe D) de 224.0.0.0 à 239.0.0.0 et un bloc d'adresses expérimentales (classe E) de 240.0.0.0 à 255.0.0.0.

Le système avec classe attribue 50% des adresses IPv4 disponibles aux 128 réseaux de classe A. 25% des adresses aux réseaux de classe B. Pour finir, la classe C partage les 25% restants avec les classes D et E.
Cette distribution a donc gaspillé de nombreuses adresses et a rapidement épuisé les adresses IPv4 disponibles. En effet, les besoins d'adresses n'étaient pas toujours couverts de manière optimale par l'une des trois classes. Ainsi, une entreprise ayant 260 hôtes devait obtenir une adresse de classe B et gaspillait de ce fait 64 740 adresses.

L'adressage avec classe a donc été abandonné au profit du système d'adressage sans classe encore utilisé aujourd'hui. Son nom formel est le routage *CIDR (Classless Inter-Domain Routing)*. L'IETF a créé un

nouvel ensemble de normes permettant d'attribuer des adresses IPv4 avec n'importe quelle limite binaire (longueur de préfixe). L'objectif était de retarder la pénurie voire l'épuisement des adresses IPv4. L'IETF n'a cependant introduit le CIDR que comme une solution temporaire et a donc commencé à chercher un successeur à l'IPv4 qui sera le futur protocole IPv6.

2.1.9 VLSM

Afin de séparer les domaines de diffusion dans un réseau, il peut être intéressant de le diviser en plusieurs sous-réseaux. Cette segmentation crée cependant une perte d'adresses réseau. En effet, les sous-réseaux contiennent chacun les mêmes nombres d'hôtes.
Si l'on regarde la figure 15 on peut constater que l'on a une perte de 28 adresses de réseau par sous-réseau WAN. La perte se monte donc à 84 adresses. On pourra résoudre ce problème à l'aide du VLSM.

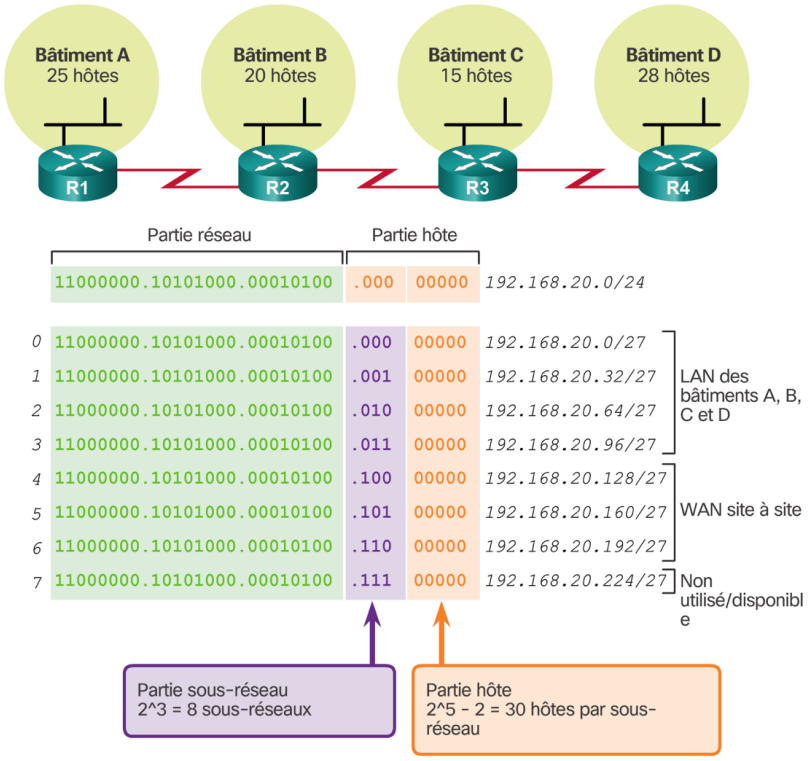


FIGURE 15 – Subnetting traditionnel

Si l'on applique maintenant le VLSM au réseau de la figure 15 on peut réduire la perte d'adresse comme le montre la figure 17.

La création de sous-réseaux VLSM est similaire à la création de sous-réseaux classiques. En effet, les formules de calcul usuelles (nombre d'hôtes, nombre de sous-réseaux) s'appliquent.
La différence réside dans la segmentation. Avec le VLSM, le réseau est divisé en sous-réseaux qui sont eux-mêmes divisés en sous-réseaux. On peut répéter ce processus plusieurs fois si l'on souhaite créer des sous-réseaux de différentes tailles (figure 16).

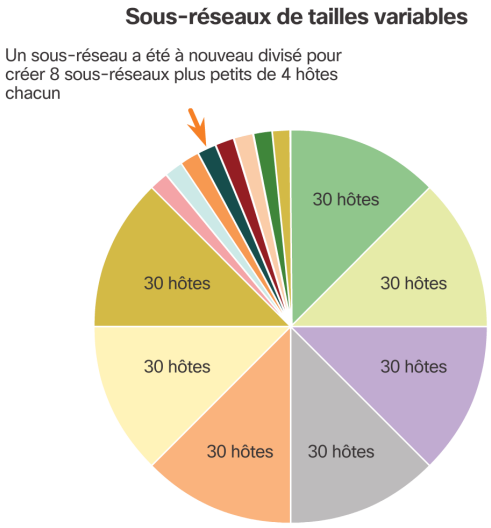


FIGURE 16 – VLSM

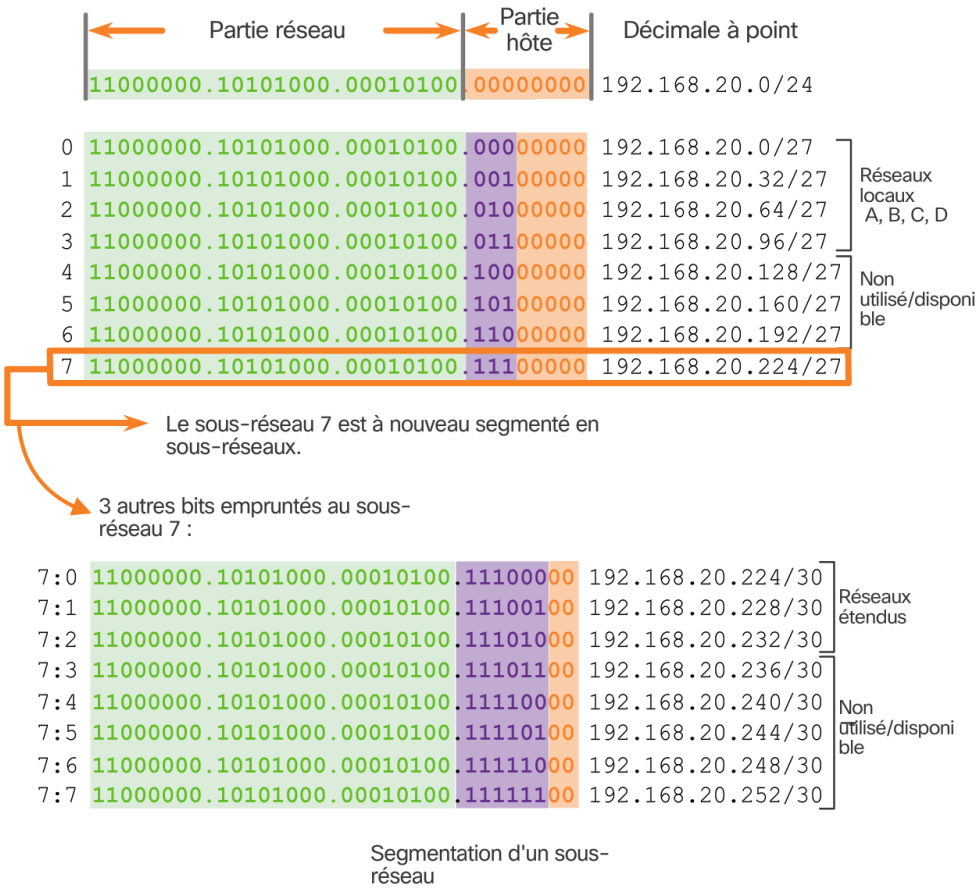


FIGURE 17 – Subnetting avec VLSM

2.2 IPv6

2.2.1 Généralités

IPv6 (Internet Protocol version 6) est le résultat des travaux menés au sein de l'IETF au cours des années 1990 pour succéder à IPv4 dont le manque d'adresses devenait problématique. Ses spécifications ont été finalisées dans la RFC 2460.
C'est un protocole réseau sans connexion de la couche 3 du modèle OSI.
Les adresses ont une taille de 128 bits au lieu des 32 bits des adresses IPv4. IPv6 dispose donc d'un espace d'adressage bien plus important qu'IPv4. En effet, il y a $2^{128} = 340$ sextillions d'adresses disponibles (340, suivi de 36 zéros).
Cette quantité d'adresses permet une grande flexibilité dans l'attribution des adresses et une meilleure agrégation des routes dans la table de routage d'Internet. Le NAT, qui a été rendu obligatoire par le manque d'adresses IPv4, n'est plus nécessaire.
Il est important de noter qu'IPv6 ne se limite pas à l'augmentation du nombre d'adresses. L'IETF a corrigé les limites de l'IPv4 lors de sa création. Par exemple, l'ICMPv6 (Internet Control Message Protocol version 6) inclut la configuration automatique et la résolution d'adresse qui étaient inexistantes dans ICMPv4

2.2.2 Coexistence des protocoles IPv4 et IPv6

La transition d'IPv4 à IPv6 ne pouvant pas être effectuée à une date fixe. IPv4 et IPv6 seront donc tous les deux utilisés en parallèle pendant un certain temps.
Il existe donc plusieurs technologies permettant à IPv4 et IPv6 de coexister. On peut notamment citer :

- **La double pile** : elle permet à l'IPv4 et à l'IPv6 de coexister sur le même segment de réseau. Les périphériques doubles pile exécutent les piles de protocoles IPv4 et IPv6 simultanément. Voir figure 18.
- **Le tunneling** : le tunneling est une méthode de transport des paquets IPv6 via un réseau IPv4. Les paquets IPv6 sont encapsulés dans des paquets IPv4, de la même manière que d'autres types de données. Voir figure 19.
- **La traduction** : les périphériques IPv6 peuvent utiliser la traduction d'adresses réseau 64 (NAT64) pour communiquer avec les périphériques IPv4. Un paquet IPv6 est traduit en un paquet IPv4, et inversement. Voir figure 20.

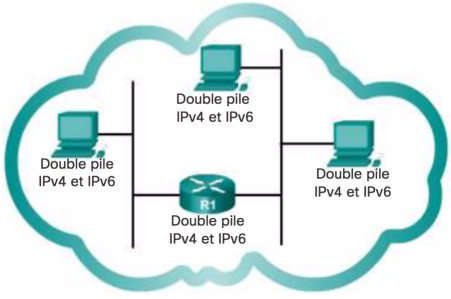


FIGURE 18 – Dual Stack

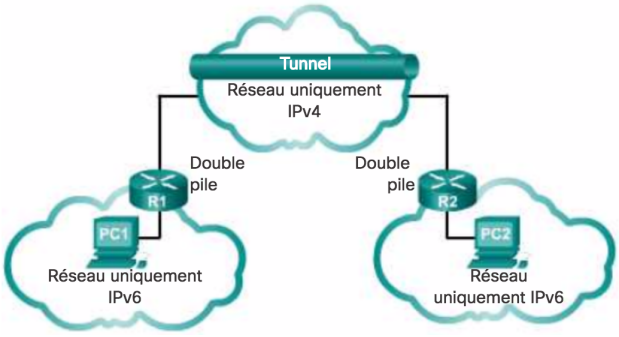


FIGURE 19 – Tunneling

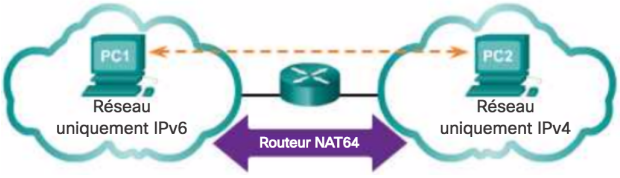


FIGURE 20 – NAT64

2.2.3 Écriture d'une adresse IPv6

Les adresses IPv6 ont une longueur de 128 bits et sont notées sous forme de chaînes de valeurs hexadécimales. Tous les groupes de 4 bits sont représentés par un caractère hexadécimal unique, de ce fait, il y aura 32 caractères hexadécimaux par adresse.
Le format privilégié pour noter une adresse IPv6 est x :x :x :x :x :x :x :x, où chaque « x » équivaut à un hextet.
Un hextet est à IPv6 ce qu'est un octet à IPv4. Un hextet est donc le terme qui désigne un segment de 16 bits ou de quatre valeurs hexadécimales. Le format privilégié implique que l'adresse IPv6 soit écrite à l'aide de 32 caractères hexadécimaux.

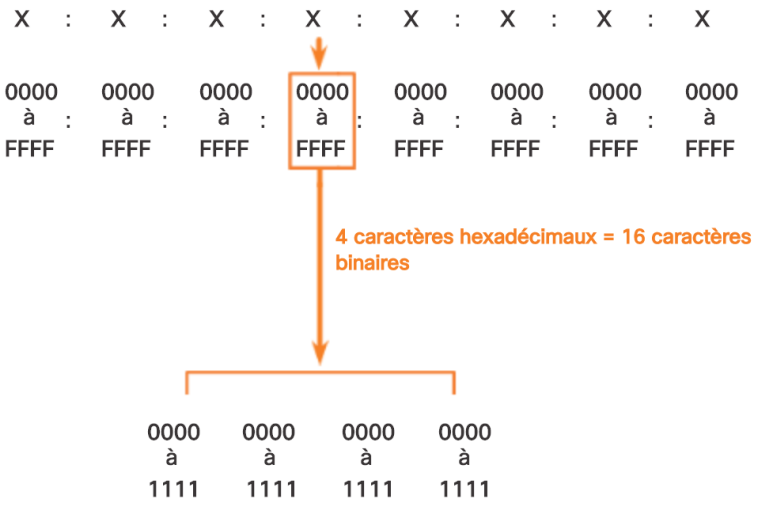


FIGURE 21 – Hextet IPv6

Il existe plusieurs règles permettant de réduire la taille d'écriture des adresses IPv6 :

— **1ère règle - Omettre les zéros en début de segment**
On peut omettre les zéros au début d'une section de 16 bits (hextet).
Attention : cette règle s'applique uniquement aux zéros du début, il ne faut jamais l'appliquer aux zéros de fin.

Recommandé	2001:0DB8:0000:1111:0000:0000:0000:0200
Sans zéros en début de segment	2001: DB8: 0:1111: 0: 0: 0: 200

FIGURE 22 – Raccourcissement selon la règle 1

— **2ème règle - Omettre les séquences composées uniquement de zéros**
Une suite de deux fois deux points (: :) peut remplacer toute chaîne unique et contiguë d'un ou plusieurs segments de 16 bits (hextets) composés uniquement de zéros.
Attention : une suite de deux fois deux points (: :) ne peut être utilisée qu'une seule fois par adresse.

Recommandé	2001:0DB8:0000:1111:0000:0000:0000:0200
Sans zéros en début de segment	2001: DB8: 0:1111: 0: 0: 0: 200
Compressé	2001:DB8:0:1111::200

FIGURE 23 – Raccourcissement selon la règle 2

2.2.4 Types d'adresses IPv6

Il existe trois types d'adresses IPv6 :

- **monodiffusion (unicast) :** une adresse de monodiffusion IPv6 identifie une interface sur un périphérique IPv6 de façon unique. Comme le montre la figure 24, une adresse IPv6 source doit être une adresse de monodiffusion.
- **multidiffusion (multicast) :** une adresse de multidiffusion IPv6 est utilisée pour envoyer un seul paquet IPv6 vers plusieurs destinations.
- **anycast :** une adresse anycast IPv6 est une adresse de monodiffusion IPv6 qui peut être attribuée à plusieurs périphériques. Un paquet envoyé à une adresse anycast est acheminé vers le périphérique le plus proche ayant cette adresse.
- **diffusion (broadcast) :** contrairement à IPv4, IPv6 n'a pas d'adresse de diffusion. Il existe cependant une adresse de multidiffusion destinée à tous les nœuds IPv6 (offre globalement les mêmes résultats).

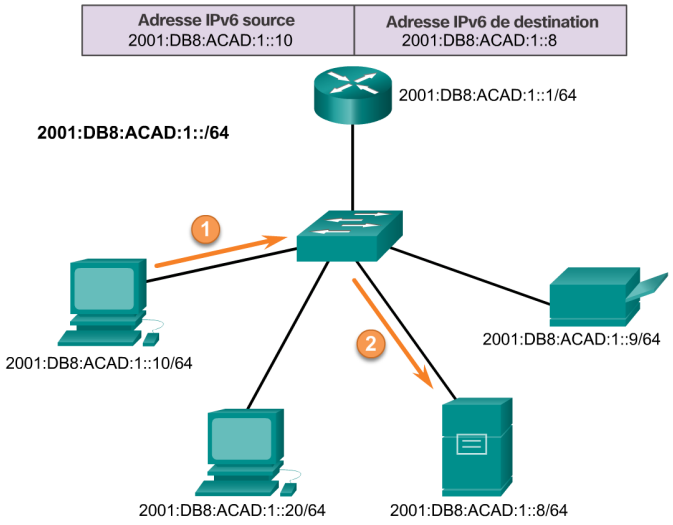


FIGURE 24 – Communication monodiffusion IPv6

2.2.5 Décomposition d'une adresse IPv6

Le protocole IPv6 n'utilise pas la notation décimale à point du masque de sous-réseau. La longueur de préfixe est utilisée pour indiquer la partie réseau d'une adresse IPv6 à l'aide de la notation *adresse IPv6/longueur de préfixe*.
La longueur de préfixe peut être comprise entre 0 et 128. La longueur de préfixe IPv6 standard est /64. Cela signifie que la partie préfixe a une longueur de 64 bits et que donc la partie ID d'interface à également 64 bits.

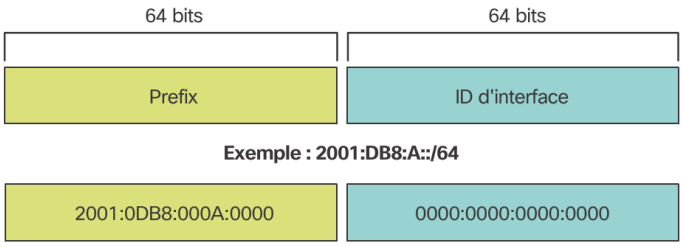


FIGURE 25 – Prefixe IPv6 standard

2.2.6 Sous-réseaux

La segmentation en sous-réseaux IPv6 exige une approche différente de celle des sous-réseaux IPv4. En effet, l'approche IPv6 fait appel à tellement d'adresses que le motif de segmentation est tout autre.
La segmentation en sous-réseaux IPv4 a pour but de limiter les domaines de diffusion et de gérer la pénurie d'adresses (VLSM).
La segmentation en sous-réseaux IPv6 n'a rien à voir avec la préservation de l'espace d'adresses. L'ID de sous réseau comprend en effet, bien plus de sous réseaux que nécessaire. La segmentation en sous-réseaux IPv6 consiste donc à créer une hiérarchie d'adressage reposant sur le nombre de sous-réseaux requis.
Parmi les deux types d'adresses IPv6 attribuables, les adresses link-local ne sont jamais segmentées en sous-réseaux, car elles n'existent que sur la liaison locale. Les adresses de monodiffusion globale IPv6 peuvent quant à elles être segmentées en sous-réseau.

L'adresse de monodiffusion globale (comme on peut le voir sur la figure 26) IPv6 se compose de :

- **Préfixe de routage global :** Partie de l'adresse attribuée par le fournisseur d'accès Internet. Généralement, les organismes d'enregistrement Internet locaux attribuent le préfixe global de routage /48 aux FAI et aux clients.
- **ID de sous réseau :** Utilisé pour identifier les sous-réseaux.
- **ID d'interface :** Équivaut à la partie hôte d'une adresse IPv4. Le terme ID d'interface est utilisé, car un hôte unique peut avoir plusieurs interfaces, chacune dotée d'une ou de plusieurs adresses IPv6.

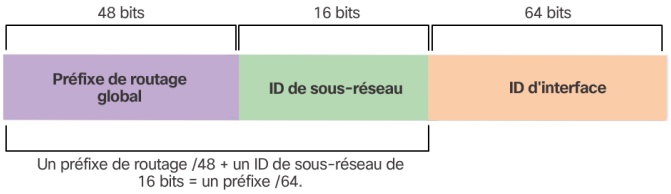


FIGURE 26 – Décomposition d'une adresse IPv6 unicast

Il est donc possible d'utiliser la partie ID de sous-réseau de 16 bits de l'adresse de monodiffusion globale IPv6 pour créer des sous-réseaux internes.
L'ID de sous réseau fournit plus de sous-réseaux et prend en charge plus d'hôtes que nécessaire :

- créer jusqu'à 65 536 sous-réseaux, sans avoir à emprunter des bits à l'ID d'interface de l'adresse.
- prendre en charge jusqu'à 18 quintillions d'adresses IPv6 d'hôte par sous-réseau (18 000 000 000 000 000 000).

Remarque : la segmentation en sous-réseaux dans l'ID d'interface à 64 bits (ou partie hôte) est également possible, mais rarement nécessaire.

FIGURE 27 – Création des sous-réseaux IPv6

3 Liaison de couche 2 - L2TPv3

3.1 Généralités

Le protocole L2TP (Layer Two Tunneling Protocol) fournit un mécanisme dynamique pour le tunneling de couche 2 à travers un réseau à commutation de paquets (par exemple, sur IP). Tel que défini à l'origine dans la RFC 2661, L2TP est une méthode standard pour le tunneling PPP (Point-to-Point Protocol). Il a depuis été adapté pour le tunneling d'un certain nombre d'autres protocoles de couche 2.
L2TPv3 permet donc l'encapsulation d'une multitude de protocoles de la couche 2 sur les réseaux IP qui sont de couche 3. Tout comme L2TP, il fournit un service pseudo-wire (émulation d'une connexion point a point).

3.2 Terminologie

Avant d'aborder le fonctionnement du protocole L2TP, il est important d'introduire certains éléments de la terminologie qui lui sont associés :

— **LCCE - L2TP Control Connection Endpoint**
Nœud L2TP qui existe à chaque extrémité d'une connexion L2TP. Peut aussi être appelé LAC ou LNS, selon que les trames tunnellées sont traitées au niveau de la liaison de données (LAC) ou de la couche réseau (LNS).

— **LAC - L2TP Access Concentrator**
Le rôle du concentrateur d'accès LAC se limite à fournir un support physique qui sera utilisé par L2TP pour transférer le trafic vers un ou plusieurs serveurs réseaux L2TP (LNS). Le concentrateur d'accès LAC joue le rôle de serveur d'accès, il est à l'origine du tunnel et est responsable de l'identification du VPN.

— **LNS - L2TP Network Server**
Les serveurs réseaux L2TP (LNS) gèrent le protocole L2TP côté serveur. Les serveurs LNS sont les émetteurs des appels sortants et les destinataires des appels entrants. Ils sont responsables de l'authentification du tunnel.

3.3 Topologies

L2TP fonctionne entre deux LCCE créant ainsi un tunnel à travers un réseau à commutation de paquets.
Il existe trois modèles de tunnels prédominants dans lesquels L2TP fonctionne :

— **LAC-LNS**
D'un côté du tunnel, le LAC reçoit le trafic d'un circuit de couche 2 et le transmet via L2TP à travers un réseau IP ou un autre réseau à commutation de paquets. De l'autre côté, un LNS termine localement le circuit de couche 2 et achemine le trafic réseau vers le réseau domestique. L'action de l'établissement de la session est pilotée par le LAC (comme appel entrant) ou le LNS (comme un appel sortant).

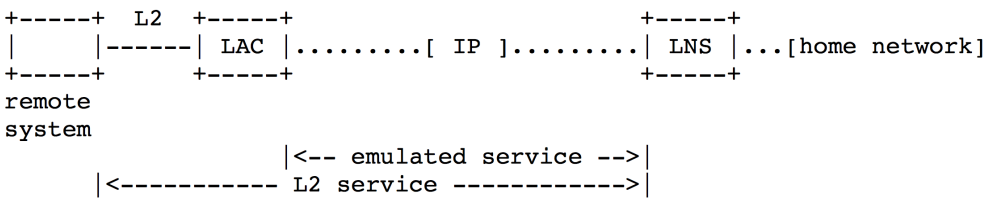


FIGURE 28 – Topologie LAC-LNS

— **LAC-LAC**
Dans ce modèle les deux LCCE sont des LAC. Chaque LAC transfère le trafic de circuit du système distant vers le LAC pair en utilisant L2TP et vice versa.
Dans sa forme la plus simple, un LAC agit comme une simple liaison croisée entre un circuit vers un système distant et une session L2TP. Ce modèle implique typiquement l'établissement symétrique : C'est-à-dire que l'un ou l'autre côté de la connexion peut initier une session à tout moment.

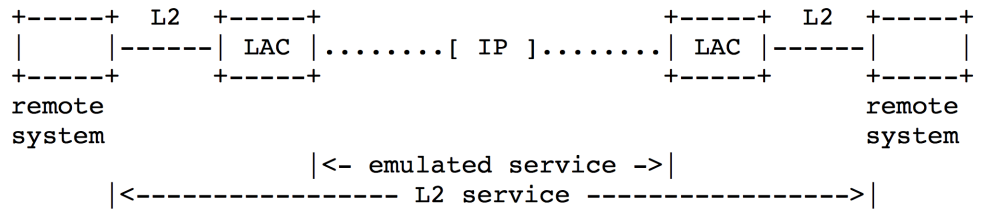


FIGURE 29 – Topologie LAC-LAC

— **LNS-LNS**
Ce modèle à deux LNS comme LCCE. Un événement au niveau de l'utilisateur, signalé ou généré par le trafic, entraîne l'établissement de la session d'un côté du tunnel. Par exemple, un tunnel généré à partir d'un PC.

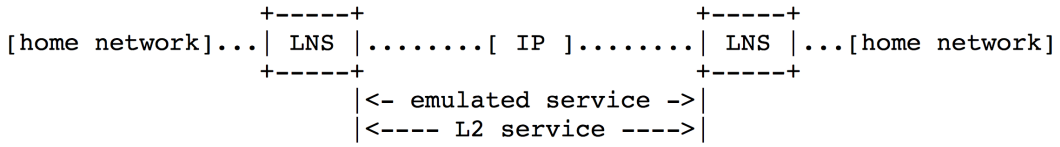


FIGURE 30 – Topologie LNS-LNS

3.4 Types de messages et en-têtes correspondants

L2TP comprend deux types de messages : les messages de type *control messages* et les messages de type *data messages* :

— Les *messages de contrôle* sont utilisés dans l'établissement, la maintenance et la suppression des connexions et des sessions. Ces messages utilisent un canal de contrôle fiable dans L2TP pour garantir leur livraison.
— Les *messages de données* sont utilisés pour encapsuler le trafic de couche 2 transporté sur la session L2TP. Contrairement aux messages de contrôle, les messages de données ne sont pas retransmis lorsqu' une perte de paquets se produit.

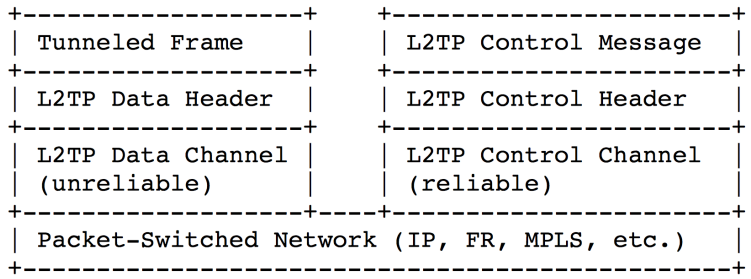


FIGURE 31 – Structure L2TPv3

Un en-tête d'un message de contrôle est défini, comme on peut le voir dans la figure 32, de la manière suivante :

— **Champ T**
Le bit T doit être à '1' afin d'indiquer qu'il s'agit d'un message de contrôle.

— **Champs L & S**
Les bits L et S doivent être à '1' afin d'indiquer que les champs d'en-tête et de numéro de séquence sont présents.

— **Champs x**
Ces bits sont réservés pour des extensions futures. Il doivent être mis à '0' dans les messages sortants et ignorés dans les messages entrants.

— **Champ Ver**
Le champ Ver permet d'indiquer la version de l'en-tête du message de contrôle L2TP.

— **Champ Length**
Le champ Length indique la longueur totale du message en octets. Il est calculé à partir du début de l'en-tête du message de contrôle (en commençant par le bit T).

— **Control Connection ID**
Le champ Control Connection ID contient l'identifiant de la connexion de contrôle. Les connexions de contrôle L2TP sont nommées par des identificateurs qui n'ont qu'une signification locale. Autrement dit, la même connexion de contrôle recevra des ID de connexion de contrôle unique par chaque LCCE.

— **Ns**
Le champ Ns indique le numéro de séquence de ce message de contrôle. Le numéro de séquence commence à 0 est est incrémenté à chaque message envoyé.

— **Nr**
Le champ Nr indique le numéro de séquence attendu dans le prochain message de contrôle à recevoir.

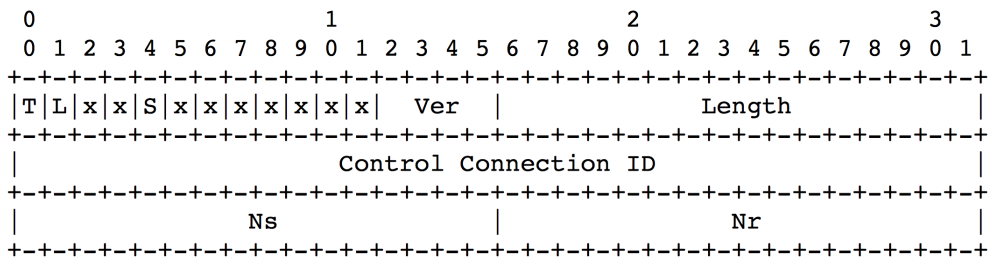


FIGURE 32 – Structure de l'en-tête d'un message de contrôle

Une en-tête d'un message de données est défini, comme on peut le voir dans la figure 33, de la manière suivante :

— **Champ L2TP Session Header**
L'en-tête de session L2TP est spécifique à l'encapsulation du réseau à commutation de paquet sur lequel le trafic L2TP est livré. L'en-tête de session doit fournir une méthode pour distinguer le trafic entre plusieurs sessions de données L2TP et une méthode pour distinguer les messages de données des messages de contrôle.

— **Champ L2-Specific Sublayer**
La sous-couche L2-Specific est une couche intermédiaire entre l'en-tête de la session L2TP et le début de la trame tunnelée. Il contient des champs de contrôle qui sont utilisés pour faciliter le tunnelage de chaque trame (par exemple, des numéros de séquence ou des flags).

— **Champs Tunnel Playload**
Charge utile du tunnel.

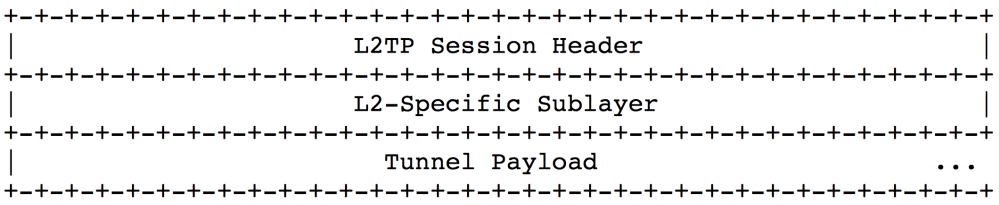


FIGURE 33 – Structure de l'en-tête d'un message de données

3.5 IPsec

L2TPv3 n'offrant pas directement l'encryption des données transitant par le tunnel, il peut être intéressant de les encrypter au préalable.
Pour ce faire, il est possible d'utiliser IPsec (Internet Protocol Security) qui est défini par l'IETF comme un cadre de standard ouvert pour assurer des communications privées et protégées sur des réseaux IP, par l'utilisation des services de sécurité cryptographiques.
IPsec opère à la couche réseau (couche 3 du modèle OSI) contrairement aux standards antérieurs qui opéraient à la couche application (couche 7 du modèle OSI). Cette particularité le rend indépendant des applications. Cela veut donc dire que les utilisateurs n'ont pas besoin de configurer chaque application aux standards IPsec pour profiter de ses avantages. Il est donc possible de sécuriser les données transitant dans le tunnel à l'aide d'une architecture L2TP over IPsec, similaire à celle présentée dans la figure 34.

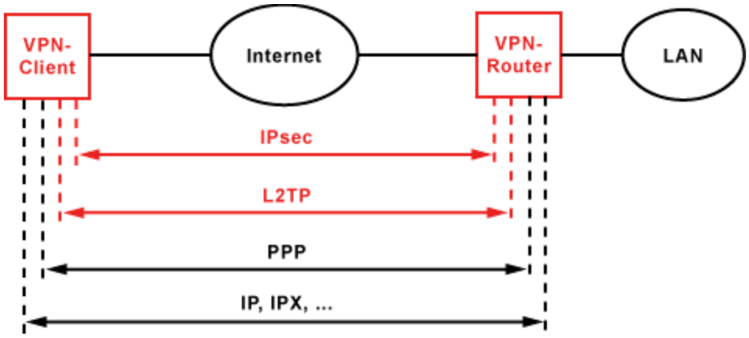


FIGURE 34 – L2TP over IPsec

3.6 Exemple de configuration

Un exemple de configuration n'utilisant pas IPSec est disponible sur le site de Cisco à l'adresse :
cisco.com
Si l'on souhaite utiliser IPSec, un autre exemple est disponible sur le site de Cisco :
cisco.com

4 Monitoring réseau

4.1 Généralités

Le monitoring réseau a pour objectif la surveillance continue d'un équipement pour s'assurer que son fonctionnement est correct. Bon nombre d'éléments peuvent être surveillés sur un équipement. Il est notamment possible de vérifier la charge CPU, l'utilisation de la mémoire, la charge réseau de/des interface(s), la température de l'équipement, etc.
La mesure de ces différents éléments permet de détecter des problèmes potentiels de manière proactive et est donc importante dans la gestion d'un réseau.
L'analyse de flux consiste à analyser le trafic circulant sur un équipement réseau pour déterminer l'utilisation du réseau. De cette manière, il est ainsi possible de connaître l'impact d'une application spécifique sur le réseau.

4.2 Définition

Un flux réseau est identifié en analysant les attributs des paquets IP. Traditionnellement, entre 5 et 7 attributs sont utilisés pour déterminer si l'empreinte du paquet est unique ou correspond à d'autres paquets :

- Adresse IP source
- Adresse IP de destination
- Port source
- Port de destination
- Type de protocole de couche 3 (IPv4, IPv6, ...)
- Type de service (streaming vidéo, voix ...)
- Interface du switch / router

L'ensemble de ces attributs sont illustrés dans la figure 35

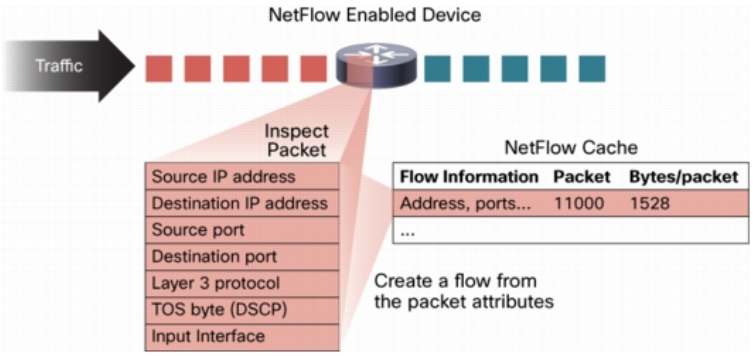


FIGURE 35 – Identification d'un flux

4.3 Applications

Une fois les différents flux de données distingués, on peut résoudre de nombreux problèmes rencontrés par les administrateurs réseaux :

- Analyser l'impact de nouvelles applications sur le réseau
- Détection de trafic non autorisé
- Validation des paramètres de QoS (Quality of Service) mis en place
- Troubleshooting d'un réseau lent
- Planification d'une extension réseau
- Détection des sources principales de trafic

4.4 Monitoring traditionnel (SNMP) vs NetFlow

Le monitoring utilisant la notion de flux (NetFlow) étend les possibilités d'audit. En effet, une solution utilisant le SNMP (Simple Network Management Protocol) n'est pas adaptée à l'analyse de flux réseau. Elle est plutôt utilisée pour contrôler l'état des équipements (CPU, mémoire, température...).
Le SNMP permet tout de même d'analyser le trafic en récupérant les compteurs de paquets ou de Bytes de chaque interface. Cependant, cette solution atteint très vite ses limites lors d'une analyse de trafic.

4.5 NetFlow

NetFlow est une architecture de surveillance des réseaux développée par Cisco. Cette fonctionnalité est intégrée à Cisco IOS. D'autres constructeurs d'équipements réseau offrent la prise en charge de NetFlow, mais sous un autre nom :

- **Jflow** - Juniper Networks
- **NetStream** - 3Com/HP et Huawei Technologies
- **Cflowd** - Alcatel-Lucent
- **Rflow** - Ericsson
- **AppFlow** - Citrix

Par ailleurs, il existe de nombreuses versions de NetFlow :

Version	Commentaire
V1	Première implémentation limitée à IPv4 sans masque réseau. Cette version est désormais obsolète
V2 à V4	Versions internes à Cisco, jamais publiées
V5	Version très répandue de NetFlow et supportée par un grand nombre de routeurs d'autres marques
V6	Version qui n'est plus prise en charge par Cisco
V7	Similaire à la version 5 mais avec un champ "routeur source"
V8	Agrégation de plusieurs informations
V9	Version qui s'appuie sur des modèles (templates), ce qui permet d'ajouter des champs sans redéfinir le standard
V10	Correspond au standard IPFIX (Internet Protocol Flow Information Export) qui est basé sur NetFlow

4.5.1 Architecture

L'architecture NetFlow est la suivante :

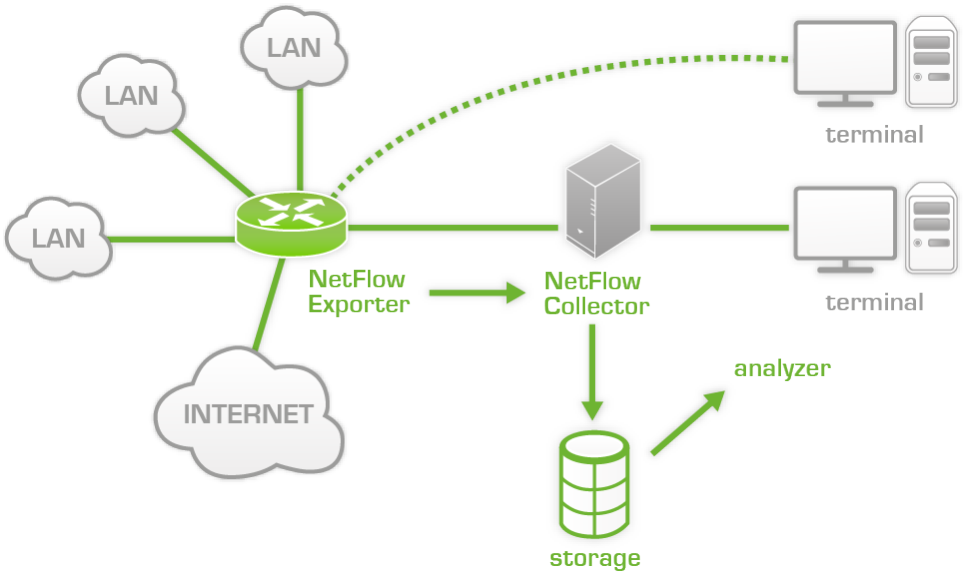


FIGURE 36 – Architecture Netflow

Cette dernière est composée de trois éléments principaux :

— **NetFlow Exporter**
Le NetFlow Exporter est un équipement réseau (routeur, switch, firewall) par lequel les données transitent. Son rôle est de déterminer le début et la fin d'un flux selon les critères définis précédemment au chapitre 4.2.
Le NetFlow Exporter utilise des timers pour déterminer si un flux est inactif. Si cette inactivité dure un certain temps, on considère le flux comme étant terminé. Un flux est également terminé lorsque le flag FIN ou RST est reçu en TCP.
Son second rôle est d'envoyer périodiquement les données des flux au NetFlow Collector.

— **NetFlow Collector**
Le NetFlow Collector est un serveur qui est placé dans un réseau afin de collecter toutes les informations de flux envoyés par les NetFlow Exporters. Le rôle du NetFlow Collector est de stocker ces informations et de fournir une interface permettant de faire des requêtes sur ces données.

— **Application d'analyse de flux**
L'application d'analyse de flux a pour objectif d'interpréter les données stockées sur le NetFlow Collector selon le contexte choisi par l'administrateur (analyse de trafic, détection d'intrusion, etc.).

4.5.2 Paquet NetFlow

Comme vu au chapitre 4.5.1, le NetFlow Exporter envoie périodiquement les données des flux détectés au NetFlow Collector à l'aide des *Export Packet* (figure 37)

FIGURE 37 – Structure d'un Export Packet NetFlow V9

Un *Export Packet* est composé d'un *Header*, d'une succession de *Template FlowSet* et de *Data FlowSet*.

- Un *FlowSet* définit une collection d'enregistrement de flux.
- Les *Data FlowSet* contiennent les données du flux.
- Les *Template FlowSet* contiennent la description des champs présents dans les futurs data FlowSet.

Le *Header* du paquet est composé, comme on peut le voir dans la figure 38 de :

- **Version** : Version des enregistrements NetFlow exportés dans le paquet (0x0009 pour la v9).
- **Count** : Nombre d'enregistrements FlowSet (template et data) contenus dans le paquet.
- **System Uptime** : Temps en ms depuis que l'équipement a démarré.
- **UNIX Seconds** : Temps en secondes depuis le 1er janvier 1970 (Coordinated Universal Time).
- **Package Sequence** : Compteur représentant le nombre d'*Export Packets* envoyés par l'équipement.
- **Source ID** : Champ de 32 bits qui identifie de manière unique les flux exportés par un équipement.

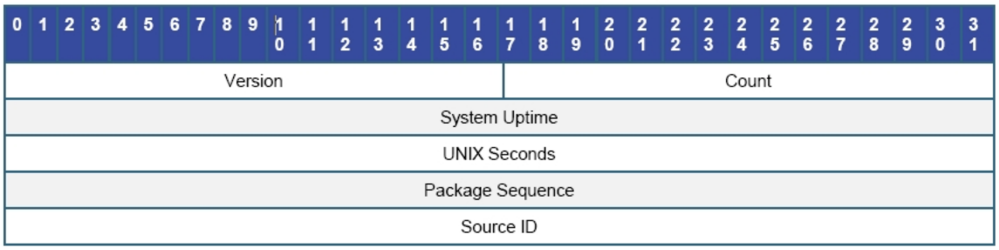


FIGURE 38 – Structure d'un Header d'un Export Packet V9

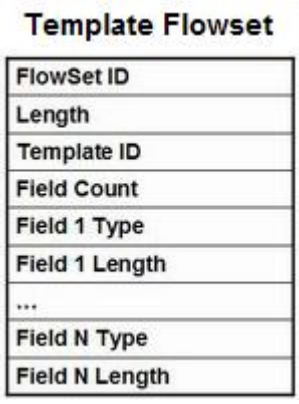


FIGURE 39 – Structure d'un Template FlowSet d'un Export Packet V9

Le *Template FlowSet* du paquet est composé, comme on peut le voir dans la figure 39 des champs :

- **FlowSet ID** : Utilisé pour distinguer les *Data FlowSet* des *Template FlowSet*. Un *Template FlowSet* a toujours un FlowSet ID compris entre 0 et 255. Un *Data FlowSet* a toujours un ID plus grand que 255.
- **Lenght** : Longueur totale du *Template FlowSet*
- **Template ID** : Identifiant unique du template
- **Field Count** : Nombre de champs dans l'enregistrement du template
- **Field Type** : Définis le type du champ. Ils sont spécifiques aux constructeurs.
- **Field Length** : Donne la longueur des champs définis précédemment, en bytes.

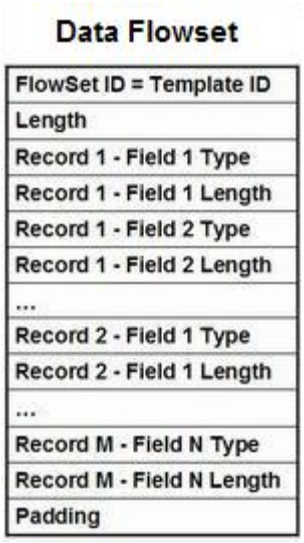


FIGURE 40 – Structure d'un Template FlowSet d'un Export Packet V9

Le *Data FlowSet* du paquet est composé, comme on peut le voir dans la figure 40 des champs :

- **FlowSet ID** : ID du FlowSet qui correspond au Template ID reçu précédemment (mapping ID-ID)
- **Lenght** : Longueur totale du *Data FlowSet*
- **Template ID** : Identifiant unique du template
- **Record N - Field N** : Sert à déterminer la valeur des champs indiqués précédemment dans le template.
- **Padding** : Inséré à la fin d'un *FlowSet* pour l'aligner sur 32 bits.

5 Conclusion

La réalisation de cette analyse m'a permis d'approfondir mes connaissances dans l'adressage IP et également de découvrir les technologies d'analyse de flux NetFlow ainsi que de tunneling L2TPv3.

Les informations contenues dans ce rapport seront très certainement utiles pour la suite du projet.

J'ai pu apprendre la matière d'une manière différente. En effet, le fait de devoir condenser une quantité importante d'informations dans un nombre de pages limité m'a contraint à sélectionner uniquement les informations utiles à la compréhension des sujets.

Ce document étant plus conséquent qu'habituellement, j'ai adopté une technique de travail qui me sera utile lors des futurs travaux de semestre.

Fribourg, le 24 mars 2017

Maic Queiroz

6 Glossaire

Abréviation	Nom complet
APIPA	Automatic Private Internet Protocol Addressing
CIDR	Classless Inter-Domain Routing
DHCP	Dynamic Host Configuration Protocol
FAI	Fournisseur d'Accès à Internet
ICANN	Internet Corporation for Assigned Names and Numbers
ICMP	Internet Control Message Protocol
IETF	Internet Engineering Task Force
IPFIX	Internet Protocol Flow Information Export
IPsec	Internet Protocol Security
IPv4	Internet Protocol version 4
IPv6	Internet Protocol version 6
L2TP(v3)	Layer 2 Tunneling Protocol (version 3)
LAC	L2TP Access Concentrator
LCCE	L2TP Control Connection Endpoint
LNS	L2TP Network Server
NAT	Network Address Translation
OSI	Open Systems Interconnection
QoS	Quality of Service
RFC	Request for Comments
RIP	Routing Information Protocol
SNMP	Simple Network Management Protocol
TCP	Transmission Control Protocol
VLSM	Variable-Length Subnet Mask
WAN	Wide Area Network

Table des figures

7 Références

7.1 Adressage IP

1. Intronet-1 2016 - CCNA R&S : Introduction to Networks
   netacad.com
2. Intronet-2 2016 - CCNA R&S : Routing and Switching Essentials
   netacad.com
3. Comment Ça Marche : adresse IP
   commentcamarche.net
4. Wikipedia : IPv4
   wikipedia.org
5. Wikipedia : IPv6
   wikipedia.org
6. Réseaux IP – F. Buntschu - 413. Adressage IPv4
   cyberlearn.hes-so.ch
7. Réseaux IP – F. Buntschu - 425. IPv6
   cyberlearn.hes-so.ch
8. RFCs :
   (a) RFC 791 - Internet Protocol
   (b) RFC 950 - Internet Standard Subnetting Procedure
   (c) RFC 1518 - An Architecture for IP Address Allocation with CIDR
   (d) RFC 1817 - CIDR and Classful Routing
   (e) RFC 2460 - Internet Protocol, Version 6 (IPv6) Specification
   (f) RFC 4632 - Classless Inter-Domain Routing (CIDR) : an Address Assignment and Aggregation Strategy

7.2 L2TP / IPSec

1. Cisco : Layer 2 Tunneling
   cisco.com
2. FrameIP : Format de l'entête L2TP
   frameip.com
3. Wikipedia : Layer 2 Tunneling Protocol Version 3
   wikipedia.org
4. Wikipedia : IPSec
   wikipedia.org
5. Elektronik Kompendium : L2TP over IPsec
   elektronik-kompendium.de
6. RFCs :
   (a) RFC 2661 - Layer Two Tunneling Protocol "L2TP"
   (b) RFC 3931 - Layer Two Tunneling Protocol - Version 3 (L2TPv3)

7.3 NetFlow

1. Wiki pandorafms : NetFlow
   wiki.pandorafms.com
2. Cisco : Introduction to Cisco IOS NetFlow
   cisco.com
3. Wikipedia : NetFlow
   wikipedia.org
4. IBM : NetFlow V9 Formats
   ibm.com
5. IGM : NetFlow
   igm.univ-mlv.fr
6. IBM : Template FlowSet field description
   ibm.com
7. Pixler : NetFlow Overview - What is a NetFlow Data Flowset ?
   plixer.com