

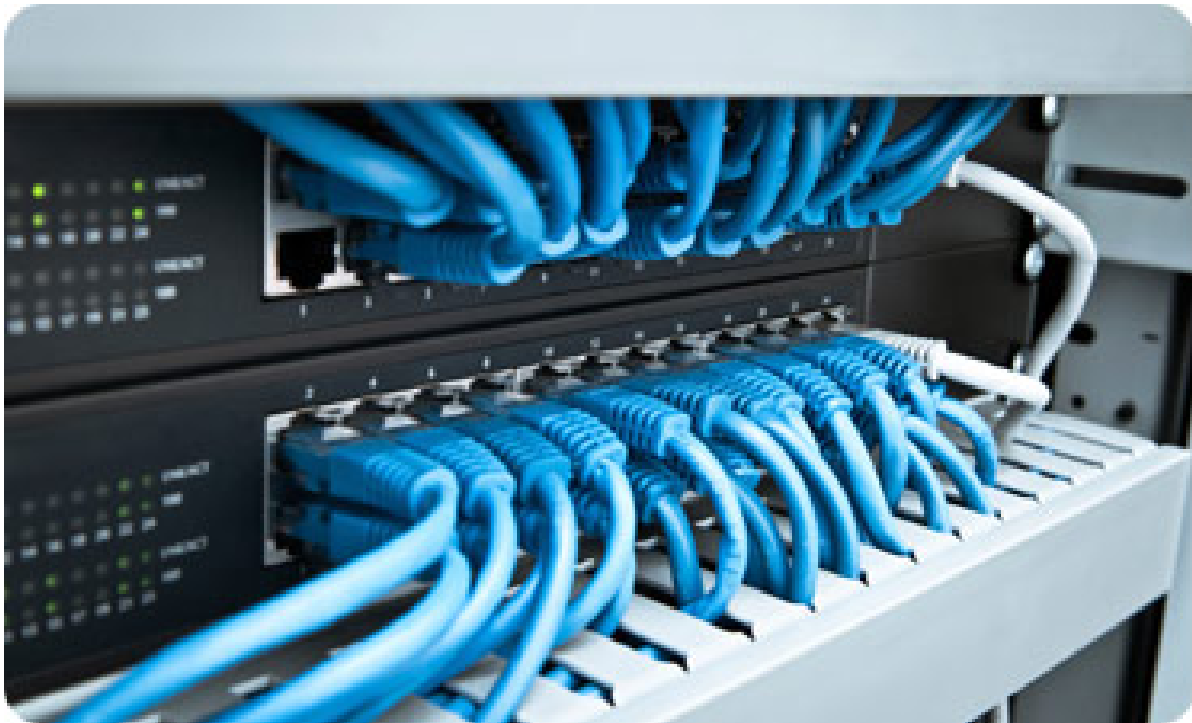


Haute école d'ingénierie et d'architecture Fribourg
Hochschule für Technik und Architektur Freiburg

Réseau IP

Révision Examen final
Partie théorie

Auteurs :
Marc ROTEN



28 juin 2018

Table des matières

1	Transition IPv4/IPv6 ==> NAT64/DNS64	5
1.1	Comment procéder	5
1.2	Double Stack	5
1.3	Tunnelling IPv6 sur IPv4	6
1.3.1	Scénario 1	7
1.3.2	Scénario 1	8
1.4	Traduction IPv4 IPv6 via NAT64/DNS64	9
2	SuperNetting (changement ISP)	11
2.1	CIDR	11
2.2	VLSM	12
3	Congestion à la source	13
3.1	Slow Start	14
4	Multicast, Quoi ?, Comment ?, Pourquoi ?	15
4.1	Multicast, c'est quoi ?	15
4.2	Pourquoi le multicast ?	15
4.3	Pourquoi pas uniquement UniCast ?	16
4.4	Comment ça marche Jamie ?	16
4.5	Désavantage	17
4.6	Use Cases	17
5	OSPF Construction des tables	18
5.1	Comment fonctionne OSPF	18
5.2	Comment connaître les voisins OSPF	18
5.3	Rappel Aires etc...	19
5.4	Les LSA et autres conneries	20
5.4.1	LSA Type 1	20

5.4.2	LSA Type 2	21
5.4.3	LSA Type 3	22
6	Fragmentation IPv6(4) + problème NAT symétrique	23
6.1	Fragmentation	23
6.2	Nat Symétrique	24
7	ICMP	25
7.1	Time Exceeded	25
7.2	Redirect	25
8	Silly Window	26
8.1	C'est quoi le silly window syndrom	26
8.2	Comment l'éviter	26
8.3	En d'autres termes	27
9	Expliquer pourquoi il y a des boucles STP	28
9.1	Nécessité du STP	28
9.2	Problèmes liés à la non-mise en place du STP	28
9.3	Résolution des problèmes	30
10	Expliquer NAT	31
10.1	Schéma conceptuel	31
10.2	Pourquoi le NAT	31
10.3	Schéma conceptuel PAT	32

Table des figures

1	Transition IPv4 vers IPv6	5
2	Transition IPv4 vers IPv6 via DOUBLE STACK	6
3	Transition IPv4 vers IPv6 via TUNNELLING	7
4	Scénario 1	7

5	Scénario 2	8
6	DNS64	9
7	DNS64	9
8	NAT64	10
9	Gaspillage adresse	11
10	Concept supernetting IPv4	11
11	VLSM	12
12	VLSM	12
13	contrôle de congestion	13
14	Schéma conceptuel Slow Start	14
15	Multicast c'est quoi ?	15
16	Multicast Pourquoi le multicast ?	15
17	Multicast Pourquoi pas uniquement UniCast ?	16
18	Multicast Comment ça marche Jamie ?	16
19	Multicast Désavantage	17
20	Multicast use cases	17
21	Relation de voisinage	18
22	Aires toussa toussa	19
23	LSA T1	20
24	LSA T2	21
25	LSA T3	22
26	Fragmentation IPv6	23
27	Fragmentation IPv6 Schéma	23
28	Nat Symétrique	24
29	Nat Symétrique	24
30	Silly Window c'est quoi ça	26
31	Silly Window c'est quoi ça	26
32	Cas nécessitant le protocole STP	28
33	Cas nécessitant le protocole STP	28



34	Cas nécessitant le protocole STP	29
35	Cas nécessitant le protocole STP	29
36	Cas nécessitant le protocole STP	30
37	NAT CONCEPT	31
38	Pourquoi le NAT	31
39	PAT CONCEPT	32

1 Transition IPv4/IPv6 ==> NAT64/DNS64

L'absence d'adresse IP en IPv4 va se poser d'ici quelques années, ou décennies, aux vues de l'expansion des appareils connectés et des technologies telles que l'IOT. il est donc nécessaire de s'inquiéter de la transition d'IPv4 vers IPv6 dès à présent.

Transition : Résoudre 2 problèmes.

1) Maintenir la connectivité vers les stations IPv4 en partageant les adresses IPv4 entre clients

Effectuer une translation entre IPv6 et IPv4

2) Fournir un mécanisme pour connecter les réseaux émergeant qui fonctionnent avec IPv6 uniquement :

Encapsulation (tunneling) des paquets IPv6 au travers de réseau uniquement IPv4

1.1 Comment procéder

- **Double pile IPv4 et IPv6**
- **Tunneling** (encapsulation) d'IPv6 dans IPv4
- **Traduction IPv6 ↔ IPv4**
 - NAT-PT, NAT64, DNS64
 - 6in4, 6to4, Teredo
 - Relais applicatifs



Processus progressif avec une longue phase de coexistence d'IPv4 et IPv6

FIGURE 1 – Transition IPv4 vers IPv6

1.2 Double 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.

Double pile IPv4 et IPv6

- Permet la compatibilité entre IPv4 et IPv6
- Une machine a des adresses IPv4 et IPv6
 - Un hôte double pile peut communiquer avec des hôtes IPv4, IPv6 et double pile
- Comment choisir la version IP d'une transmission ?
 - DNS répond à une requête avec une adresse IPv4, IPv6 ou les deux

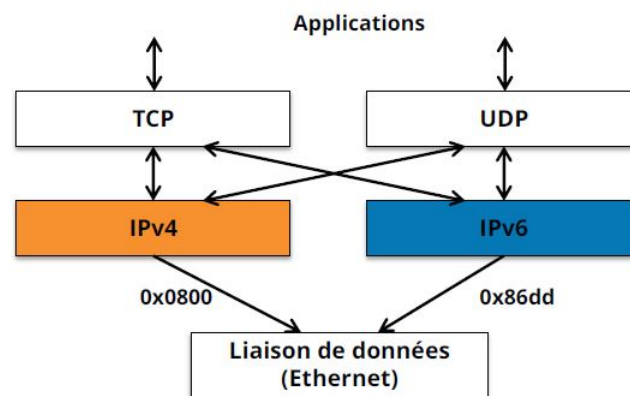


FIGURE 2 – Transition IPv4 vers IPv6 via DOUBLE STACK

1.3 Tunnelling IPv6 sur IPv4

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.

Tunnels IPv6 sur IPv4

- Les réseaux de transit seront migrés progressivement vers IPv6
- L'infrastructure IPv4 restera en place (Routeurs double pile)
- Transmission des datagrammes IPv6 des clients sur l'infrastructure IPv4

Scénario I

- Des îlots IPv6 sont interconnectés à travers un réseau IPv4
- Un **tunnel configuré** IPv6 sur IPv4 est établi **entre routeurs**

Scénario II

- Des hôtes IPv6 établissent **automatiquement** un tunnel vers leur destination à travers un réseau IPv4
- requiert des routeurs à double pile, avec le support d'adresses dédiées et les adresses IPv6 'compatibles' IPv4. Par exemple: 2002::160.98.100.125

FIGURE 3 – Transition IPv4 vers IPv6 via TUNNELLING

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

1.3.1 Scénario 1

Tunnels IPv6 sur IPv4 : Scénario I

L'interconnexion de stations et réseaux IPv4 et IPv6 n'est pas triviale. A côté d'une possibilité « *dual stack* » (communications séparées), on utilise en général du « *tunneling* » en encapsulant de l'IPv6 dans des paquets IPv4.

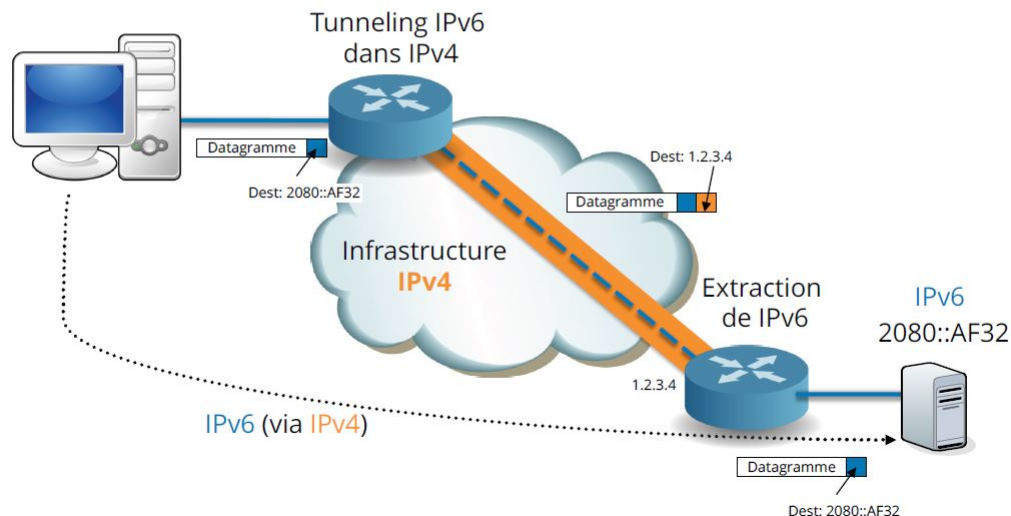


FIGURE 4 – Scénario 1

1.3.2 Scénario 1

Tunnels IPv6 sur IPv4 : Scénario II

- Utilisable si la terminaison du tunnel est une **station** et permet d'établir un tunnel **sans configuration**
- Utilise une adresse IPv6 « compatible IPv4 » au format : `2002::d.d.d.d`

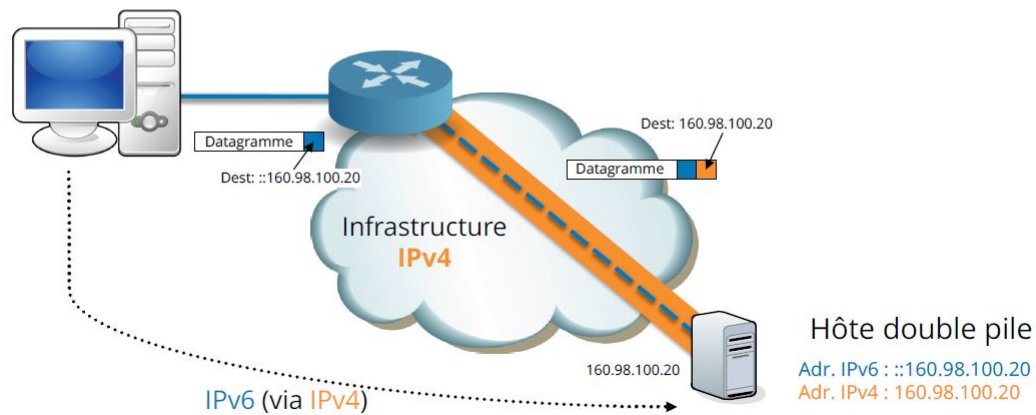


FIGURE 5 – Scénario 2

1.4 Traduction IPv4 IPv6 via NAT64/DNS64

NAT64/DNS64 (1) (RFC 6146 & 6147)

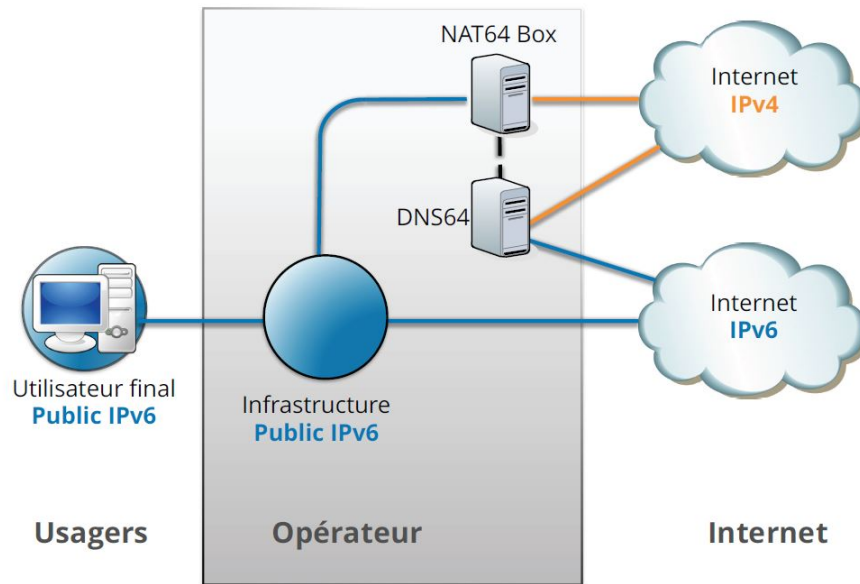


FIGURE 6 – DNS64

NAT64/DNS64 (2)

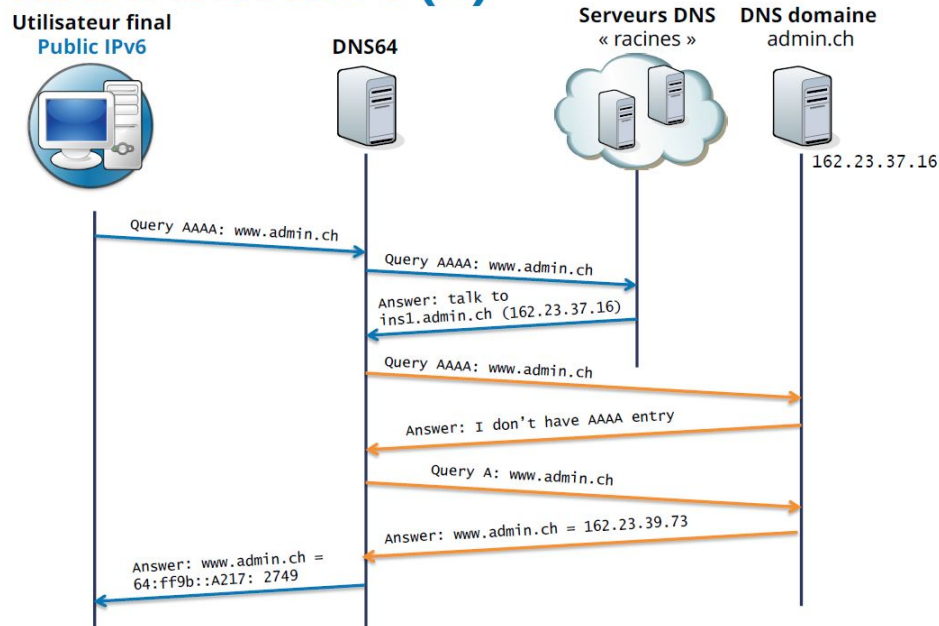


FIGURE 7 – DNS64

NAT64/DNS64 (3)

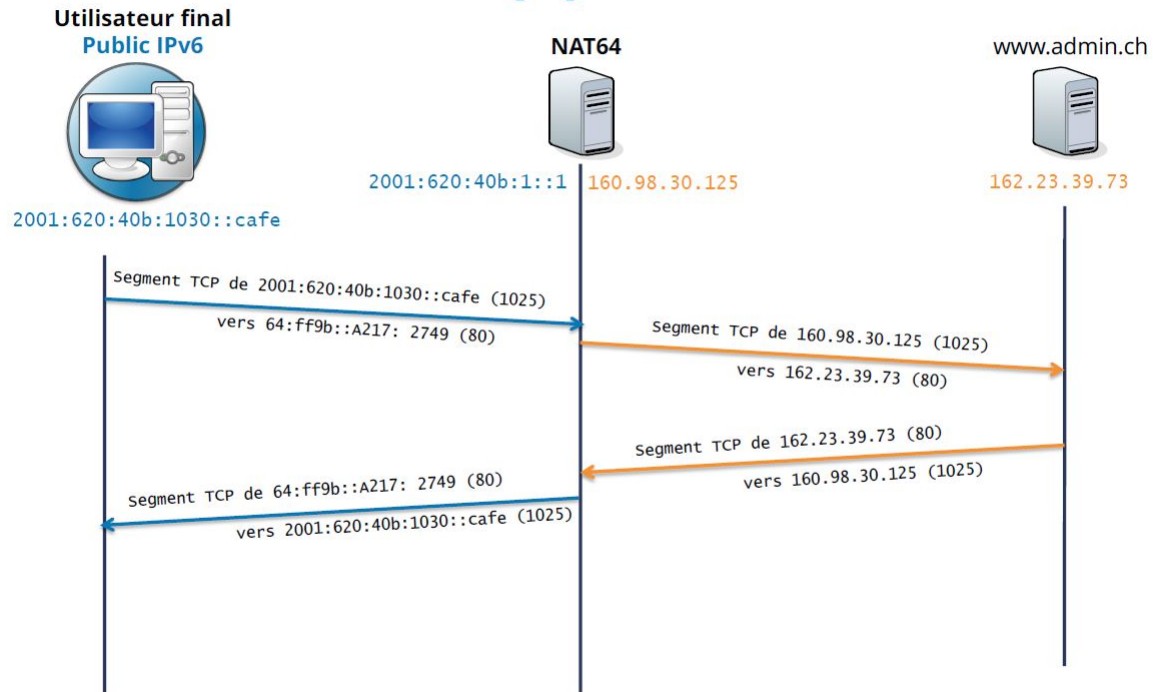


FIGURE 8 – NAT64

2 SuperNetting (changement ISP)

2.1 CIDR

Mise en place du Classless Inter-Domain Routing (CIDR) pour éviter le gaspillage d'adresse comme le montre le schéma ci dessous

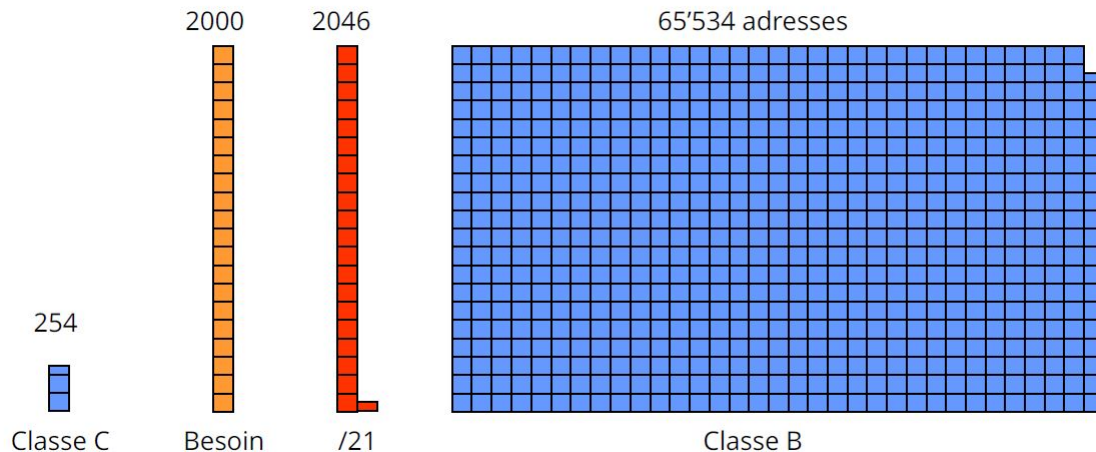


FIGURE 9 – Gaspillage adresse

Supernetting

Supernetting, par opposition à *subnetting*, étend le concept de classes et *subnets* de façon à permettre à des organisations ou ISPs de **grouper** des adresses réseau de classes C en utilisant des bits de poids faibles de l'adresse réseau comme des bits de stations/*subnets*.

Exemple avec 1024 (1022) adresses de classe C obtenues en groupant 4 adresses réseau de classe C:

194.200.0	11000010.11001000.00000000. xxxxxxxx
194.200.1	11000010.11001000.00000001. xxxxxxxx
194.200.2	11000010.11001000.00000010. xxxxxxxx
194.200.3	11000010.11001000.00000011. xxxxxxxx
Masque	11111111.11111111.11111111. 00.00000000

La notation du *supernet* obtenu sera la suivante: 194.200.0.0/22

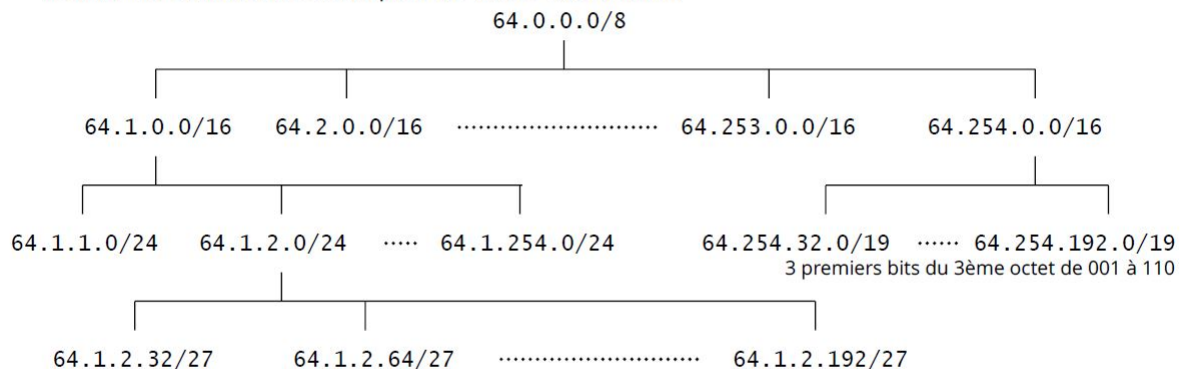
10 bits à disposition pour les adresses de stations et de subnets

FIGURE 10 – Concept supernetting IPv4

2.2 VLSM

VLSM (Variable Length Subnet Mask)

VLSM (RFC 1812, RFC 2644) permet d'optimiser la répartition d'adresses à l'intérieur d'un réseau en organisant des *subnets* hiérarchiques de taille différentes sous la même adresse réseau en utilisant des masques de tailles différentes.



Note: Le protocole de routage doit transmettre les masques dans les annonces si on utilise le VLSM.

FIGURE 11 – VLSM

Les routeurs supportant le VLSM doivent trouver les adresses de sous-réseaux dans leur table de routage en utilisant la corrélation la plus longue possible avec l'adresse de destination dans la représentation binaire. C'est ce qu'on appelle le "longest prefix match".

Table de routage		
Route No.	Adresse IP/masque	Adresse IP
1	64.1.2.0/24	<u>01000000.00000001.00000010.00000000</u>
2	64.1.0.0/16	<u>01000000.00000001.00000000.00000000</u>
3	64.0.0.0/8	<u>01000000.00000000.00000000.00000000</u>

Destination 64.1.2.5 01000000.00000001.00000010.00000101

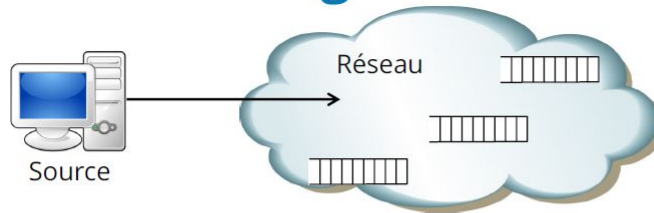
La route 1 est choisie sur le critère du *longest prefix match*.

FIGURE 12 – VLSM

3 Congestion à la source

Lors de connexion TCP, on veut que le débit soit maximal, il est donc nécessaire d'y aller progressivement, jusqu'à atteindre le meilleur compromis.

Contrôle de congestion TCP à la source



Contrôle à la source: la station source envoie un certain nombre d'octets selon une **fenêtre de congestion** (*congestion window*) puis attend de recevoir un acquittement avant de continuer (fenêtre coulissante). La taille de cette fenêtre est donnée en MSS (*Maximal Segment Size*, taille maximale des segments négociée à l'établissement).

→ **Implémentations:**

- **Slow start:** au début de chaque connexion, la station source augmente progressivement la taille de la fenêtre de congestion en partant de 1 MSS. L'augmentation est de 1 MSS par acquittement reçu ce qui conduit à une augmentation à peu près exponentielle de cette fenêtre si le délai d'acquittement est négligeable. Au-delà d'un **seuil de congestion**, la progression est ralentie dans la zone de prévention des congestions (partie linéaire)
- **Réponse aux congestions:** quand une congestion est détectée par la source (*timeout* d'un acquittement), la source réduit volontairement la valeur du seuil de congestion à la moitié de la valeur actuelle de la fenêtre de congestion (mais ≥ 2 MSS) et recommence le *slow start*.

FIGURE 13 – contrôle de congestion

3.1 Slow Start

Slow Start

Visualisation de l'augmentation « exponentielle » de la taille de la fenêtre de congestion

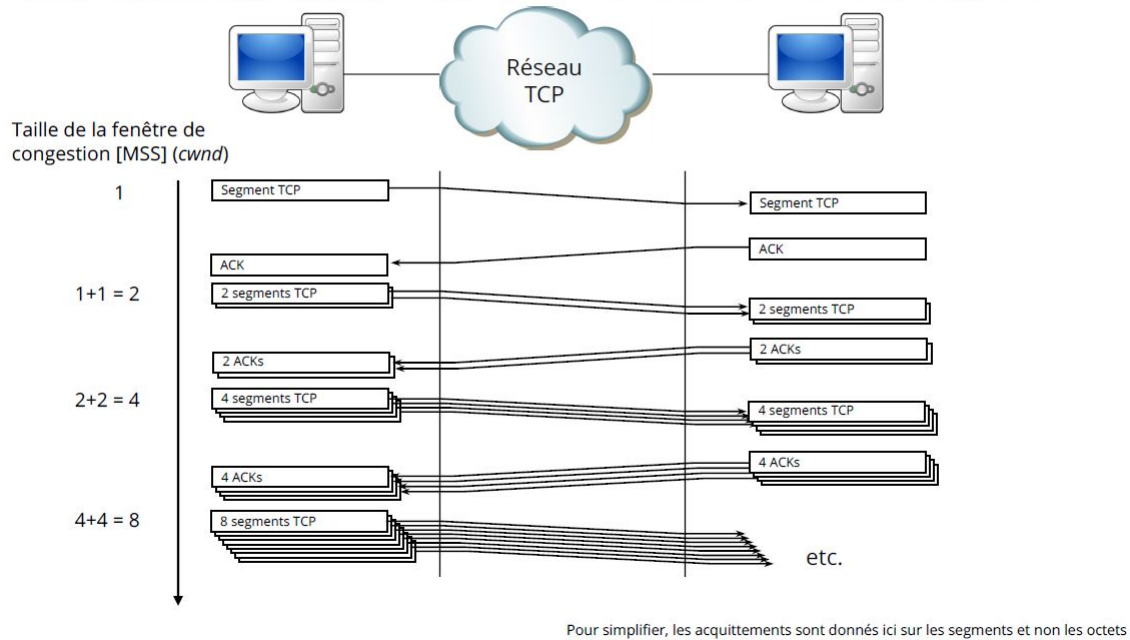


FIGURE 14 – Schéma conceptuel Slow Start

4 Multicast, Quoi ?, Comment ?, Pourquoi ?

Le multicast Est basé sur des échanges en UDP. Pas de nécessité d'établir une connexion. On parle de protocole best-effort.

4.1 Multicast, c'est quoi ?

» Multicast is data transmission to a group of destinations simultaneously

- I.e one to many transmission

FIGURE 15 – Multicast c'est quoi ?

4.2 Pourquoi le multicast ?

» Main goal of multicast is reduce the load on...

- Sending server processing
- Network bandwidth resources
- Router forwarding processing
- Receiving host processing

FIGURE 16 – Multicast Pourquoi le multicast ?

4.3 Pourquoi pas uniquement UniCast ?

Why Not Just Use Unicast?

- » Sender must generate one packet for each receiver
 - Called “head-end replication”
- » Sender must know addresses of all receivers
- » Routers must process packets for each receiver separately
- » Bandwidth use is proportional to number of receivers

FIGURE 17 – Multicast Pourquoi pas uniquement UniCast ?

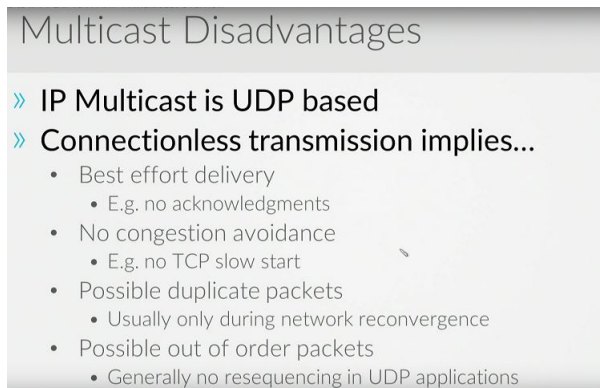
4.4 Comment ça marche Jamie ?

Routers make a single forwarding decision for all recipients

- Only one packet is replicated per interface, saving bandwidth
- Uninterested hosts do not receive packets

FIGURE 18 – Multicast Comment ça marche Jamie ?

4.5 Désavantage




Multicast Disadvantages

- » IP Multicast is UDP based
- » Connectionless transmission implies...
 - Best effort delivery
 - E.g. no acknowledgments
 - No congestion avoidance
 - E.g. no TCP slow start
 - Possible duplicate packets
 - Usually only during network reconvergence
 - Possible out of order packets
 - Generally no resequencing in UDP applications

FIGURE 19 – Multicast Désavantage

4.6 Use Cases



Multicast Use Case Examples

- » **Multimedia**
 - IPTV & IP Video Surveillance
 - Videoconferencing
 - VoIP Music on Hold
- » **Data distribution**
 - Large scale data replication
- » **Real-time applications**
 - Stock tickers

FIGURE 20 – Multicast use cases

5 OSPF Construction des tables

5.1 Comment fonctionne OSPF

Les routeurs exécutant OSPF doivent établir des relations de voisinage (neighbor adjacency) avant d'échanger des routes. Comme OSPF est un protocole de routage d'état de liaison (Link State), les voisins n'échangent pas de tables de routage. Au lieu de cela, ils échangent des informations sur la topologie du réseau. Chaque routeur OSPF exécute ensuite l'algorithme SFP (Shortest Path First – Chemin le plus court) pour calculer les meilleures routes et les ajoute à la table de routage. Étant donné que chaque routeur connaît la topologie complète d'un réseau, la probabilité d'une boucle de routage est minime.

Chaque routeur OSPF stocke les informations de routage et de topologie dans trois tables :

==>Table de voisins (Neighbor table) – stocke des informations sur les voisins OSPF

==>Table de topologie (Topology table) – stocke la structure de topologie d'un réseau

==>Table de routage (Routing table) – stocke les meilleurs itinéraires

5.2 Comment connaître les voisins OSPF

es routeurs OSPF doivent établir une relation de voisinage avant d'échanger des mises à jour de routage. Les voisins OSPF sont dynamiquement découverts en envoyant des paquets Hello sur chaque interface OSPF sur un routeur. Les paquets Hello sont envoyés à l'adresse IP de multidiffusion 224.0.0.5.

Le processus est expliqué dans la figure suivante :

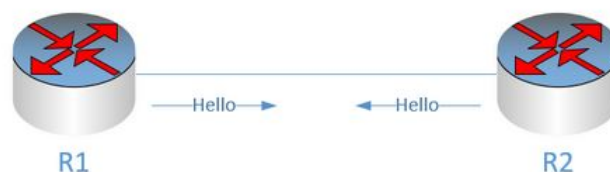


FIGURE 21 – Relation de voisinage

Les routeurs R1 et R2 sont directement connectés. Une fois OSPF activé, les deux routeurs envoient des paquets Hello les uns aux autres pour établir une relation de voisinage.

5.3 Rappel Aires etc...

OSPF utilise le concept de zones (area). Une zone est un regroupement logique de réseaux et de routeurs contigus. Tous les routeurs dans la même zone ont la même table de topologie, mais ils ne connaissent pas les routeurs dans les autres zones. Le principal avantage de la création de zones est que la taille de la topologie et la table de routage d'un routeur sont réduites, qu'il faut moins de temps pour exécuter l'algorithme SPF et que les mises à jour de routage sont également réduites.

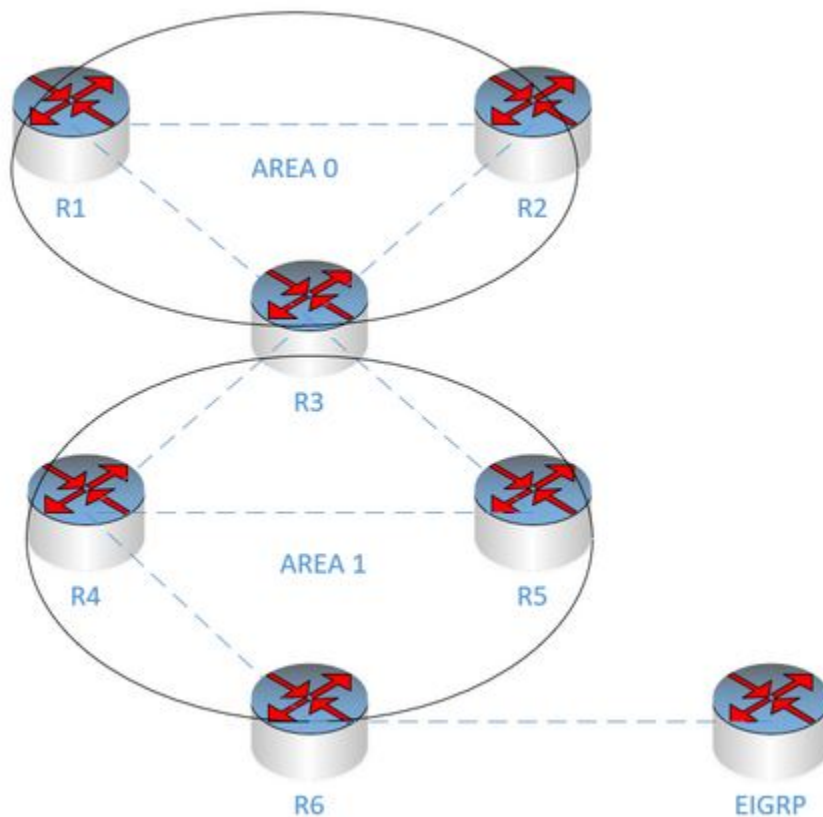


FIGURE 22 – Aires toussa toussa

Chaque zone du réseau OSPF doit se connecter à la zone de backbone (area 0). Tous les routeurs à l'intérieur d'une zone doivent avoir le même ID de zone pour devenir des voisins OSPF. Un routeur qui a des interfaces dans plus d'une zone (zone 0 et zone 1, par exemple) s'appelle Area Border Router (ABR). Un routeur qui connecte un réseau OSPF à d'autres domaines de routage (réseau EIGRP, par exemple) s'appelle Autonomous System Border Routers (ASBR).

Tous les routeurs exécutent OSPF. Les routeurs R1 et R2 sont à l'intérieur de la zone backbone (zone 0).

Le routeur R3 est un ABR car il possède des interfaces dans deux zones différentes, à savoir la zone 0 et la zone 1.

Le routeur R4 et R5 sont dans la zone 1.

Le routeur R6 est un ASBR, car il connecte le réseau OSPF à un autre domaine de routage. (EIGRP dans ce cas).

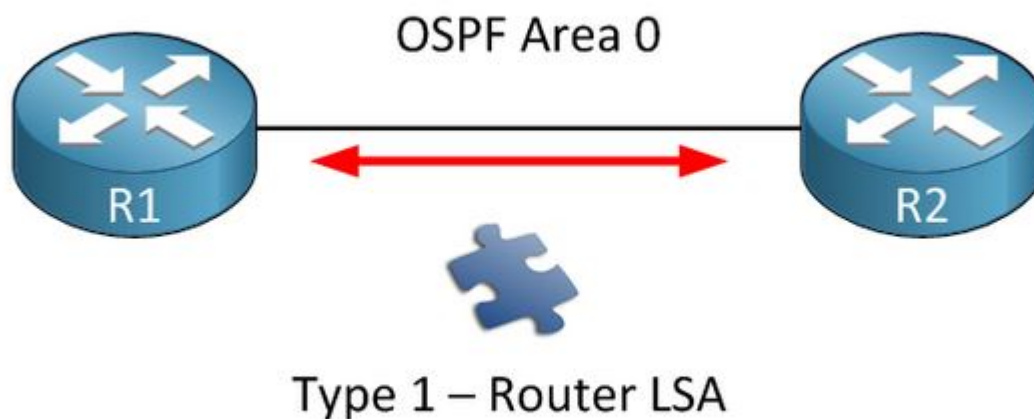
Si le sous-réseau directement connecté du R1 échoue, le routeur R1 envoie la mise à jour de routage uniquement à R2 et R3, car toutes les mises à jour de routage sont toutes localisées dans la zone.

5.4 Les LSA et autres conneries

Les annonces LSA (Link-State Advertisements) sont utilisées par les routeurs OSPF pour échanger des informations de topologie.

5.4.1 LSA Type 1

Here's the first LSA Type:



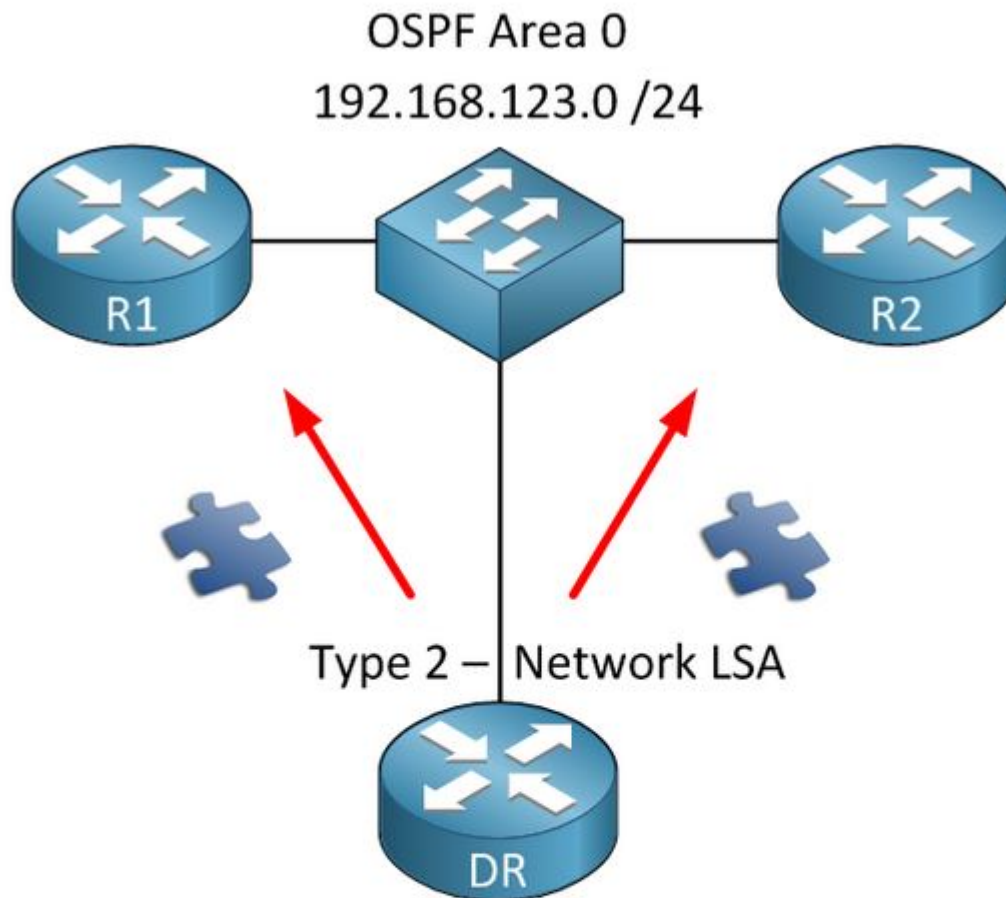
Each router within the area will flood a **type 1 router LSA** within the area. In this LSA you will find a list with all the directly connected links of this router. How do we identify a link?

FIGURE 23 – LSA T1

Tous les routeurs vont spammer à l'intérieur de leur zone des LSA type 1 pour élire le DR, le BDR etc...

5.4.2 LSA Type 2

The second LSA type (network LSA) is created for multi-access networks:



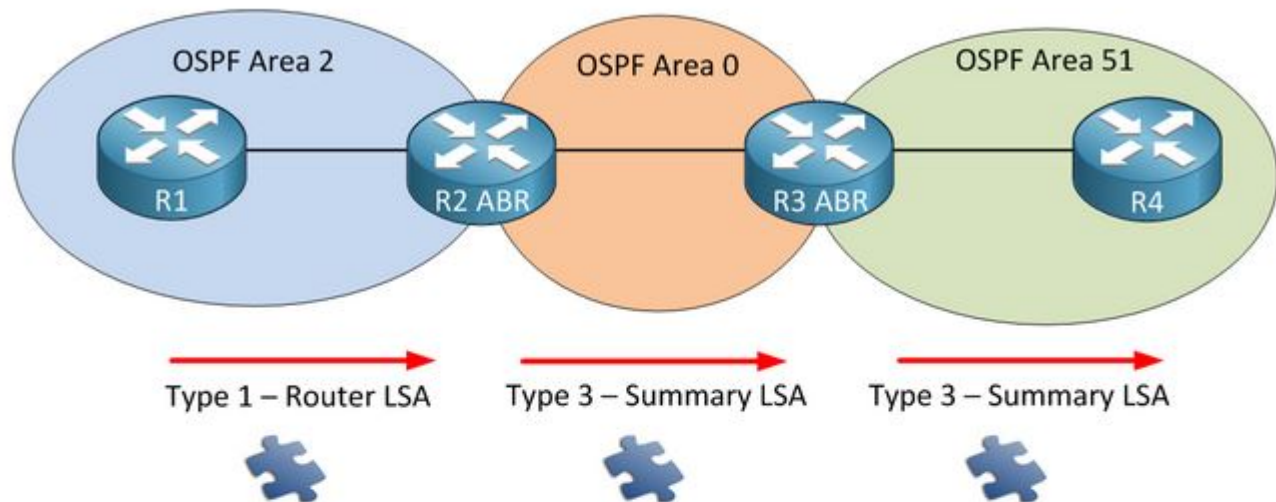
The **network LSA** or **type 2** is created for each multi-access network. Remember the OSPF network types? The **broadcast** and **non-broadcast** network types require a DR/BDR. If this is the case you will see these network LSAs being generated by the DR. In this LSA we will find all the routers that are connected to the multi-access network, the DR and of course the prefix and subnet mask.

In my example above we will find R1, R2 and the DR in the network LSA. We will also see the prefix 192.168.123.0 /24 in this LSA. Last thing to mention: the network LSA always **stays within the area**.

FIGURE 24 – LSA T2

5.4.3 LSA Type 3

Let's look at the third LSA type:



Type 1 router LSAs **always stay within the area**. OSPF however works with multiple areas and you probably want full connectivity within all of the areas. R1 is flooding a router LSA within the area so R2 will store this in its LSDB. R3 and R4 also need to know about the networks in Area 2.

R2 is going to create a **Type 3 summary LSA** and flood it into area 0. This LSA will flood into all the other areas of our OSPF network. This way all the routers in other areas will know about the prefixes from **other areas**.

The name “summary” LSA is very misleading. By default OSPF is **not going to summarize** anything for you. There is however a command that let you summarize inter-area routes. Take a look at my [OSPF summarization tutorial](#) if you are interested. If you are looking at the routing table of an OSPF router and see some **O IA** entries you are looking at LSA type 3 summary LSAs. Those are your inter-area prefixes!

FIGURE 25 – LSA T3

FIGURE 27 – Fragmentation IPv6 Schéma

6.2 Nat Symétrique

NAT symétrique (1)

NAT symétrique: Une nouvelle adresse publique ou un nouveau numéro de port source sont utilisés pour chaque destination. La différence avec le Port restricted NAT est qu'une entrée dans la table de translation est créée pour chaque communication.

- Seul la machine externe (son adresse IP et son port) pour laquelle le paquet est destiné est autorisée à dialoguer avec la machine interne (son adresse IP et son port) qui a initié la connexion.
- Algorithme complexe à implémenter
- Augmentation du niveau de sécurité
- Un numéro de port est « consommé » par flux actif, ce qui limite le nombre de flux simultanés à 65535, par adresse IP publique.

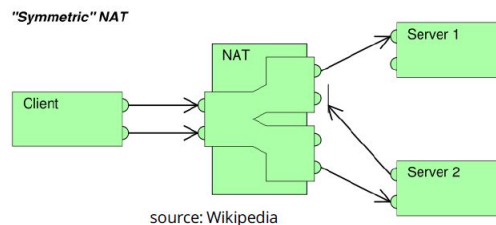


FIGURE 28 – Nat Symétrique

La table de routage du routeur gérant le Nat en mode symétrique est donc sensiblement différent. Comme illustré ci-dessous.

NAT symétrique (2)

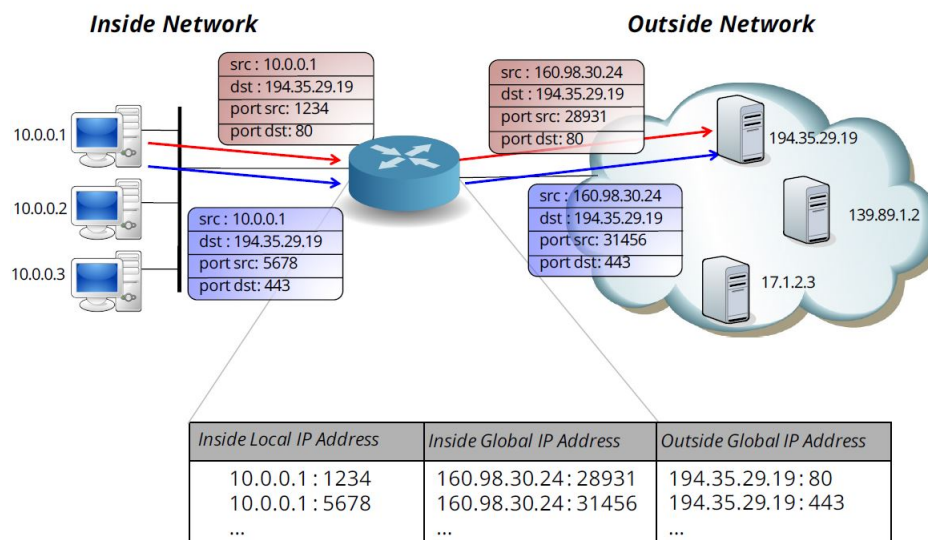


FIGURE 29 – Nat Symétrique

7 ICMP

7.1 Time Exceeded

Signification : ce message est envoyé lorsque le temps de vie d'un datagramme ou le temps de réassemblage des parties d'un datagramme est dépassé. L'en-tête du datagramme est renvoyé pour que l'utilisateur sache quel datagramme a été détruit.

Ce message ICMP est envoyé quand le TTL arrive à zéro. il est envoyé quand on dispose de plusieurs routes d'un point A vers un point B. et qu'un paquet se perd. Le TTL arrive à zéro et le datagramme est détruit.

7.2 Redirect

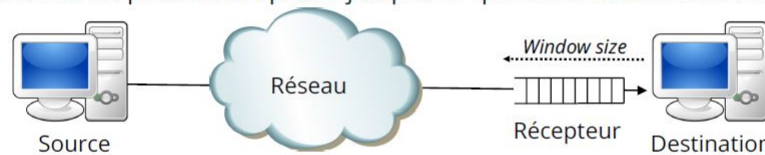
Ce message est envoyé lorsque le chemin pris par un datagramme à l'intérieur de notre réseau utilise un chemin non optimal. Un ICMP redirect est envoyé à la source pour changer justement la route employée pour joindre le point distant.

8 Silly Window

8.1 C'est quoi le silly window syndrom

TCP n'impose pas de longueur **minimale** aux segments TCP. Mais si ceux-ci sont trop petits, la transmission sur le réseau devient inefficace car la proportion du trafic due aux en-têtes TCP et IP est trop grande.

Imaginons ce qui se passe si à un moment donné l'application à une extrémité d'une connexion TCP n'arrive pas à absorber tous les octets qui arrivent à la mémoire tampon du récepteur TCP. Celui-ci va diminuer la fenêtre de réception et l'indiquer à la source du trafic TCP, probablement à plusieurs reprises jusqu'à ce que cette fenêtre soit mise à zéro.



Ensuite, dès que quelques octets pourront passer vers l'application, le récepteur va ouvrir la fenêtre. La source va toute suite envoyer un segment de cette taille si elle a des octets à envoyer. La mémoire tampon au récepteur va de nouveau saturer et la fenêtre pour la source sera de nouveau mise à zéro.

Le résultat sera une succession de petits segments, à la limite avec un octet de donnée, qui vont passer de la source au récepteur TCP. C'est le « *Silly Window Syndrome* » (syndrome de la fenêtre stupide). Le trafic sera donc très inefficace sur ce réseau.

FIGURE 30 – Silly Window c'est quoi ça

8.2 Comment l'éviter

TCP évite le « *Silly Window Syndrome* » en prenant deux mesures:

1. La destination ne modifie pas la taille fenêtre de réception avec de petites corrections. Si cette taille est modifiée (en plus ou en moins), la **différence est au moins d'un MSS** (ou de la moitié de la taille de la mémoire tampon de réception si cette valeur est inférieure au MSS).
2. Si la source a envoyé des données qui n'ont **pas encore été quittancées**, les données **suivantes** sont mises dans une mémoire tampon de transmission. Elles ne peuvent être envoyées que si une des trois conditions suivantes est vérifiée:
 - (i) les octets envoyés précédemment sont quittancés par la destination ou
 - (ii) la source a accumulé assez de données pour envoyer un segment d'un MSS ou
 - (iii) la source a accumulé au moins autant de données que la moitié de la taille de la plus grande fenêtre reçue pendant cette connexion.C'est l'**algorithme de Nagle** (RFC 896). Notons que si les segments précédents ont été quittancés, la source peut très bien envoyer un petit segment.

FIGURE 31 – Silly Window c'est quoi ça

8.3 En d'autres termes

Le silly window size c'est en quelque sorte un réseau qui n'est pas efficace, car une application génère des segment TCP trop petit, du coup la destination réduit sa Window Size. Et dès que le réseau n'est plus congestionné, la Window size n'est donc pas assez rapide pour absorber tout le trafic

Pour pallier ce problème, la destination ne fait pas d'infimes corrections sur sa window size. Toutes les modifications apportées doivent être d'au moins 1 MSS.

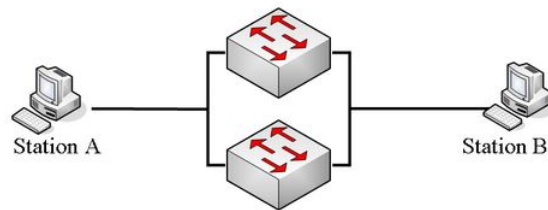
Et si des données n'ont pas encore été acquitées, elles sont mises en mémoire. Voir ci-dessus les conditions nécessaires!!

9 Expliquer pourquoi il y a des boucles STP

9.1 Nécessité du STP

Architecture redondée

Maintenant que l'on souhaite que les paquets entre les ordinateurs A et B transitent même en cas de panne matériel, créons cette nouvelle architecture:



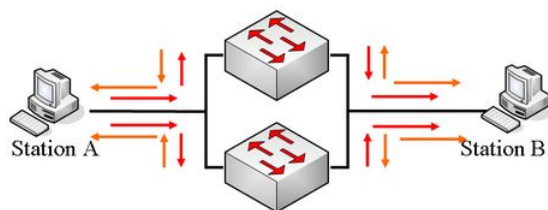
Avec cette architecture, on voit bien que si le switch du haut ne fonctionne plus, le switch du bas peut tout même transmettre les paquets de A vers B et de B vers A.

FIGURE 32 – Cas nécessitant le protocole STP

9.2 Problèmes liés à la non-mise en place du STP

1er problème: Tempête de broadcast

Sur l'architecture redondée précédente, imaginons que la station A envoie un message de broadcast (trame niveau 2 avec comme adresse MAC de destination FFFF.FFFF.FFFF). Que se passe-t-il?



- Le switch du haut reçoit la trame sur son port, **extraie l'adresse MAC** de destination (FFFF.FFFF.FFFF) et **la duplique sur tous ses ports** car c'est une adresse de broadcast. La trame sort donc du switch du haut et se dirige vers le switch du bas
- idem pour le switch du bas; il reçoit la trame sur son port, **extraie l'adresse MAC** de destination (FFFF.FFFF.FFFF) et **la duplique sur tous ses ports** car c'est une adresse de broadcast. La trame sort donc du switch du bas et se dirige vers le switch du haut
- et ces trames **tournent sans arrêt** entre les 2 switches, faisant monter leur CPU à 100% et les font plus ou moins planter (souvent un reboot est nécessaire)

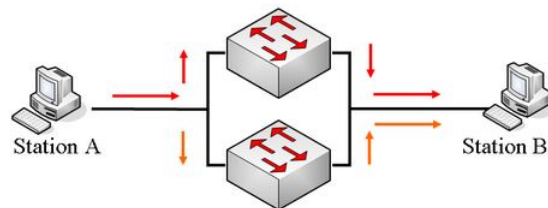
Ce phénomène s'appelle la **tempête de broadcast**, ou **broadcast storm** en anglais.

FIGURE 33 – Cas nécessitant le protocole STP

2ème problème: Duplication de trame

Maintenant, imaginons que la station A envoie une trame vers la station B, donc la trame sera forgée avec les informations suivantes:

- adresse MAC source: A
- adresse MAC destination: B



Que se passe-t-il?

- Le switch du haut reçoit la trame sur son port (flèche rouge), **extraie l'adresse MAC** de destination (B) et la commute sur le port de droite. La station B reçoit bien la trame de la station A
- Mais le switch du bas reçoit aussi la trame sur son port (flèche orange), **extraie l'adresse MAC** de destination (B) et la commute sur le port de droite. La station B reçoit donc pour une deuxième fois la trame de la station A

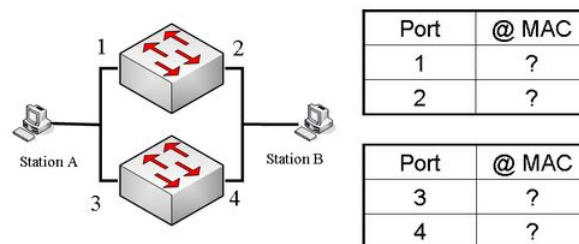
Ce phénomène s'appelle la **duplication de trame** (pas top comme optimisation réseau 😊)

FIGURE 34 – Cas nécessitant le protocole STP

3ème problème: Instabilité de la table CAM

Maintenant, regardons un peu ce qu'il se passe côté table CAM – Content Addressable Memory – du switch.

Pour ceux qui ont oublié cette notion, je vous renvoi vers ce chapitre ([switch](#)).



Reprenons la trame précédente (message de A vers B):

- la trame arrive sur le port 1 du switch du haut. Le switch **extraie l'adresse MAC source** et **l'insère dans sa table CAM** [port 1 = adresse MAC A]
- la trame arrive aussi sur le port 3 du switch du bas. Le switch **extraie l'adresse MAC source** et **l'insère dans sa table CAM** [port 3 = adresse MAC A]

FIGURE 35 – Cas nécessitant le protocole STP

9.3 Résolution des problèmes

Résolution des 3 problèmes

Pour éviter ces 3 problèmes (**tempête de broadcast**, **duplication de trame** et **instabilité de la table CAM**), le protocole spanning-tree a été créé. Comme ces problèmes proviennent du fait que le réseau commuté est face à une boucle physique, le spanning-tree permet d'identifier cette boucle et de la bloquer "logiciellement".

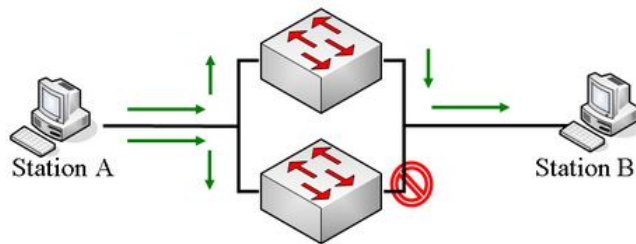


FIGURE 36 – Cas nécessitant le protocole STP

10 Expliquer NAT

10.1 Schéma conceptuel

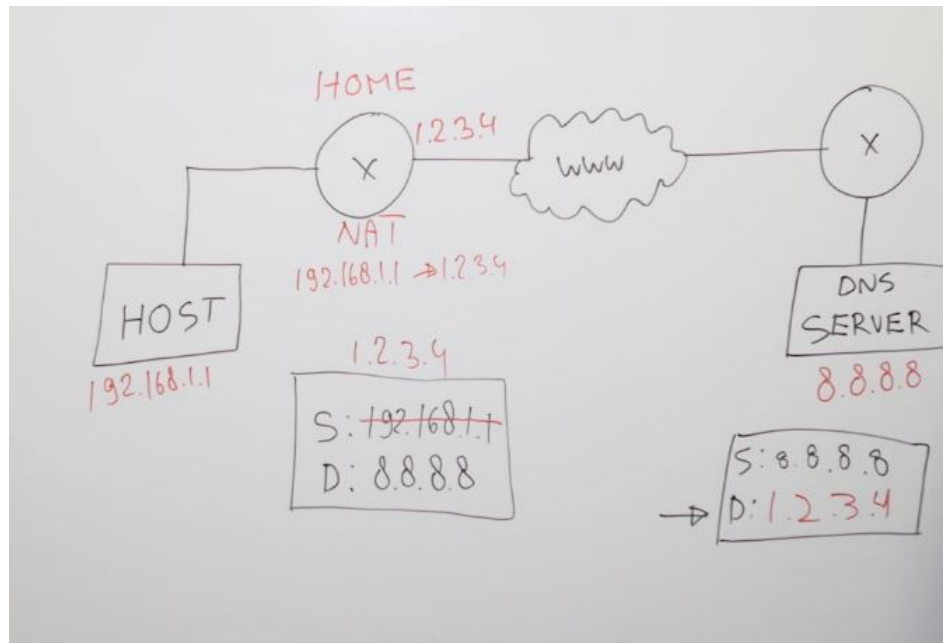


FIGURE 37 – NAT CONCEPT

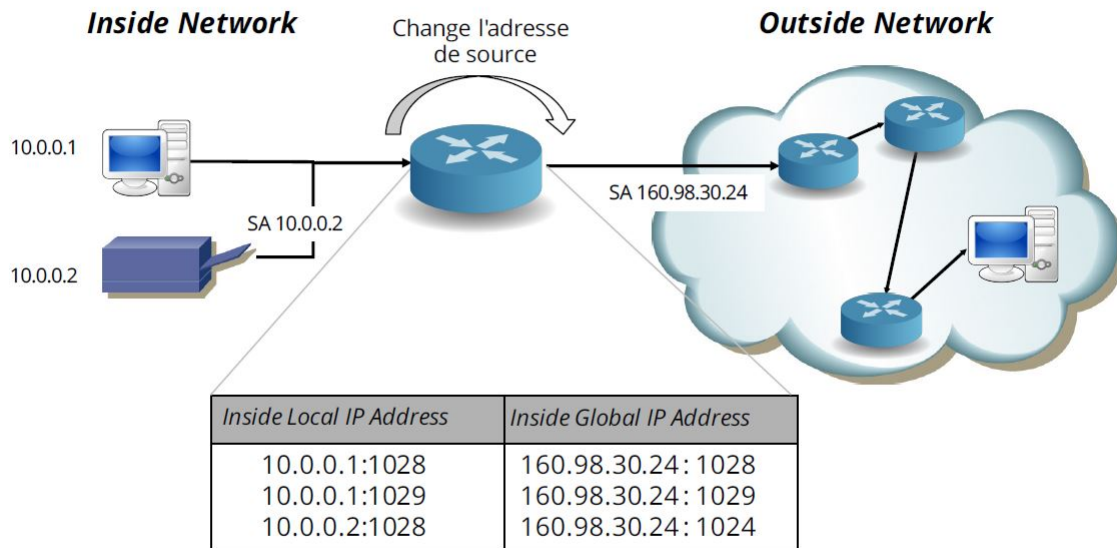
10.2 Pourquoi le NAT

- Considérations de **sécurité**: cache les adresses "inside" au monde extérieur
- **Économise** des adresses IP
- Permet à un réseau d'accéder à Internet sans devoir **enregistrer** toutes les adresses de sous-réseaux auprès de l'autorité d'attribution d'adresses Internet
- Permet de connecter deux réseaux qui ont des **adresses identiques**
- Permet de garder un groupe d'adresse après un changement d'ISP

FIGURE 38 – Pourquoi le NAT

10.3 Schéma conceptuel PAT

Port Address Translation (PAT)



Toutes les stations "internes" utilisent la même adresse IP vu de l'extérieur, multiplexage avec le *Port Number*. PAT utilise en premier le port de source (si pas déjà utilisé pour une autre station source). PAT libère l'entrée dans la table de translation quand le message « FIN » de TCP est observé sur cette connexion.

FIGURE 39 – PAT CONCEPT