Fondamentaux

Fais-toi une raison, il faut passer par la théorie avant la pratique garçon !

- [Fondamentaux] TCP
- [Fondamentaux] Chiffrement symétrique / asymétrique
- [Fondamentaux] ARP / NDP
- [Fondamentaux] IPv6

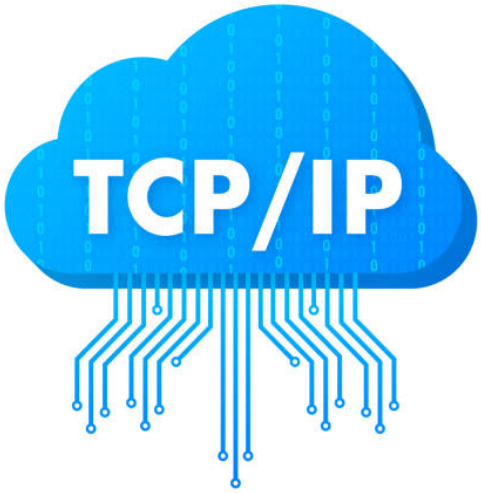
[Fondamentaux] TCP

Introduction

Le **TCP** pour *Transmission Control Protocol* est un protocole réseau de la couche *Transport* du **modèle OSI.**

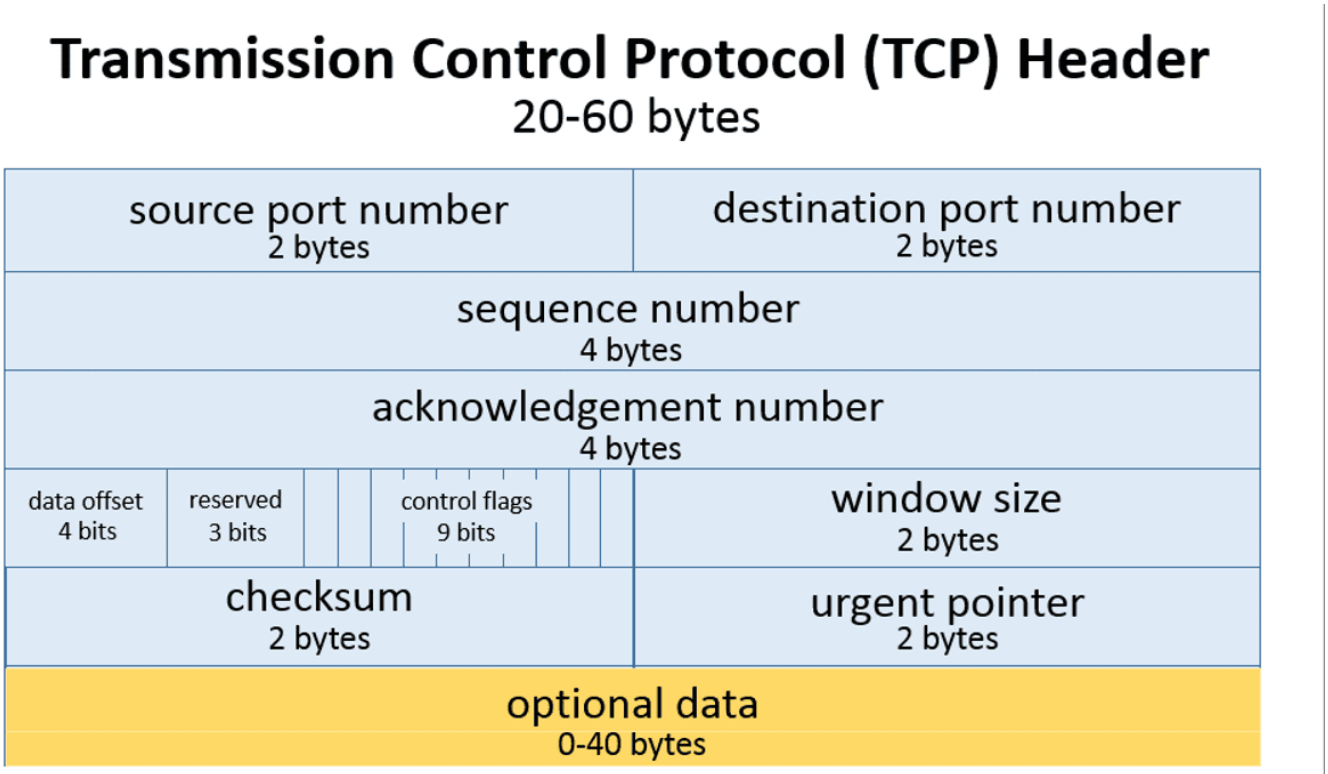
Il est réputé pour sa **fiabilité** puisqu'il assure **l'intégrité** des paquets.

Par conséquent, il est aussi plus lent que l'UDP.



Composition

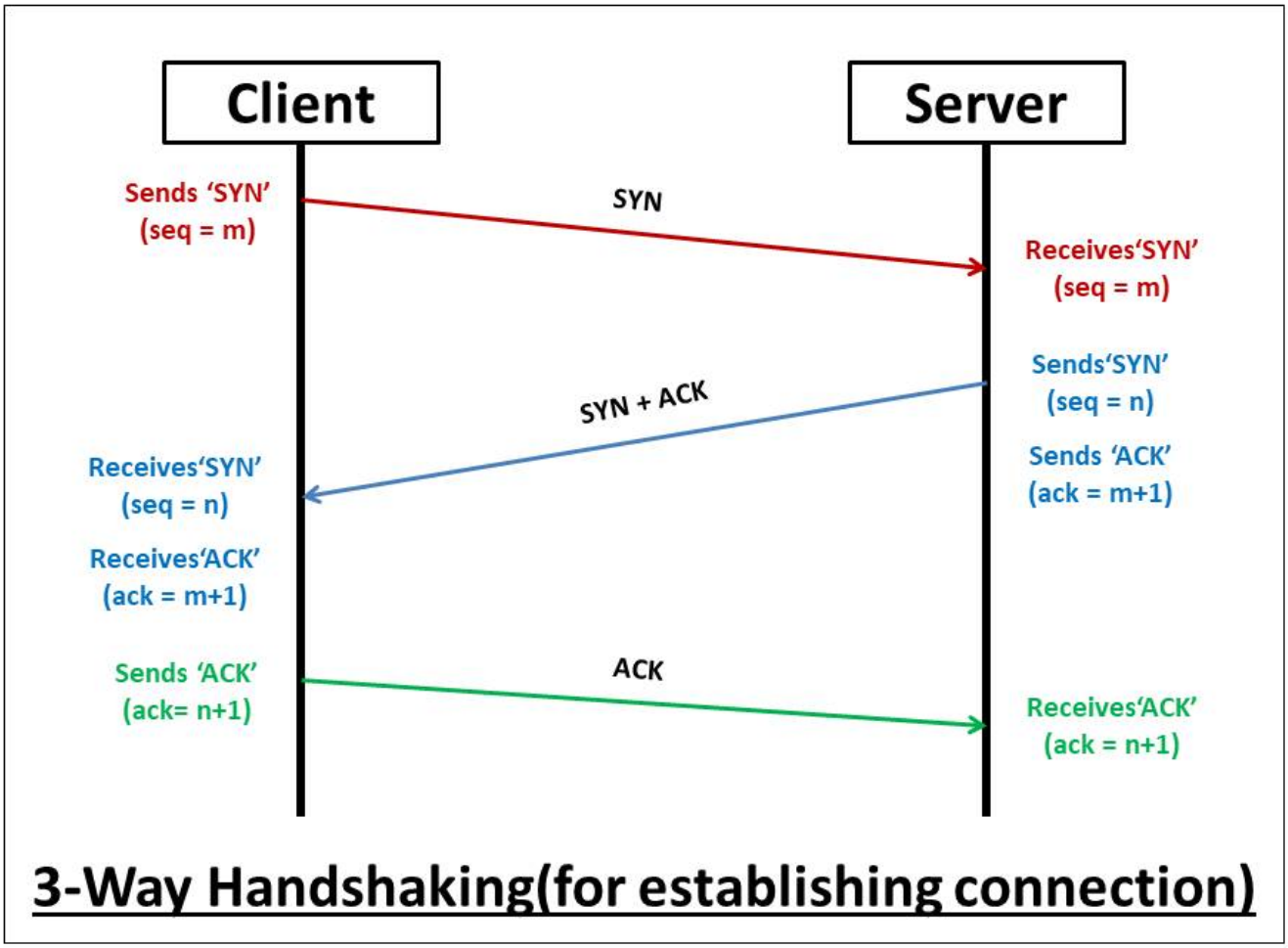
Voici à quoi ressemble le protocole TCP :



- Le **sequence number** est un nombre aléatoire fixé par le système d'exploitation.
- Le **acknowledgment number** qui sera égal au numéro de séquence reçue, incrémentée de un.
- Le **window size** qui sert à définir le nombre de paquet à recevoir avant d'envoyer un *ack* en réponse.
- Le **checksum** qui est la somme de contrôle pour vérifier l'intégrité du segment.

Three-way Handshake

Il s'agit de la procédure lors de l'établissement d'une connexion TCP standard qui se passe en trois étapes (d'où le nom).

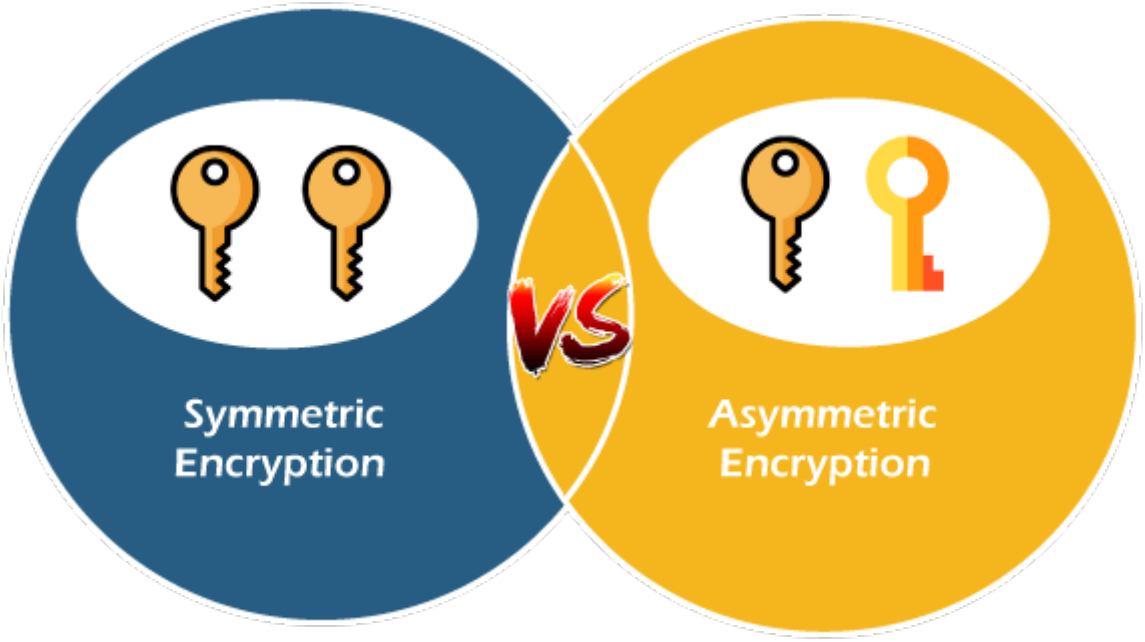


3-Way Handshaking(for establishing connection)

[Fondamentaux] Chiffrement symétrique / asymétrique

Introduction

Cette page va décrire les fonctionnements des deux grands types de chiffrement que sont chiffrement **symétrique** et **asymétrique**.



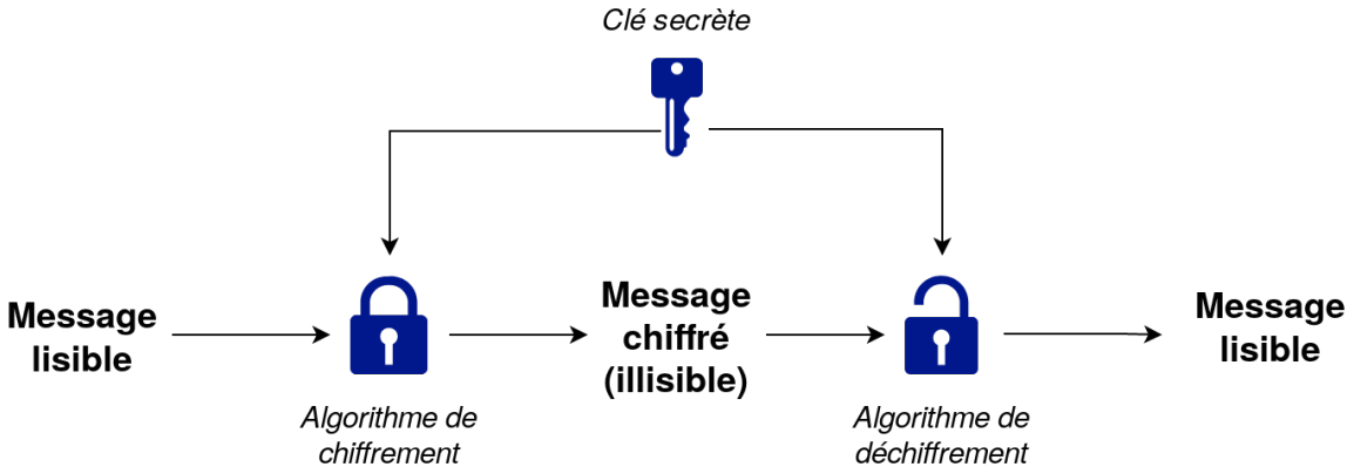
Chiffrement symétrique

Il est le type de chiffrement le **moins sécurisé** mais est encore utilisé notamment pour sa **consommation de ressource qui est moindre** que celle du chiffrement asymétrique.

Le fonctionnement est assez simple : la donnée d'entrée va être chiffrée avec une **clé** par un **algorithme de chiffrement** pour obtenir une donnée incompréhensible en sortie qui pourra transiter sans risque de **confidentialité**.

Ensuite, le récepteur va récupérer la donnée chiffrée qu'il va pouvoir décoder avec la **même clé** qui a servit à chiffrer cette donnée et avec le même algorithme de chiffrement.

Exemple : Le chiffrement **César** qui prend une phrase en entrée et qui va décaler chaque lettre d'un certains nombre.



Chiffrement asymétrique

Réputé pour son **niveau de sécurité plus robuste**, il est utilisé dans la plupart des protocoles et applications sécurisées.

Il est cependant plus lent et gourmand en ressource.

Voici son fonctionnement :

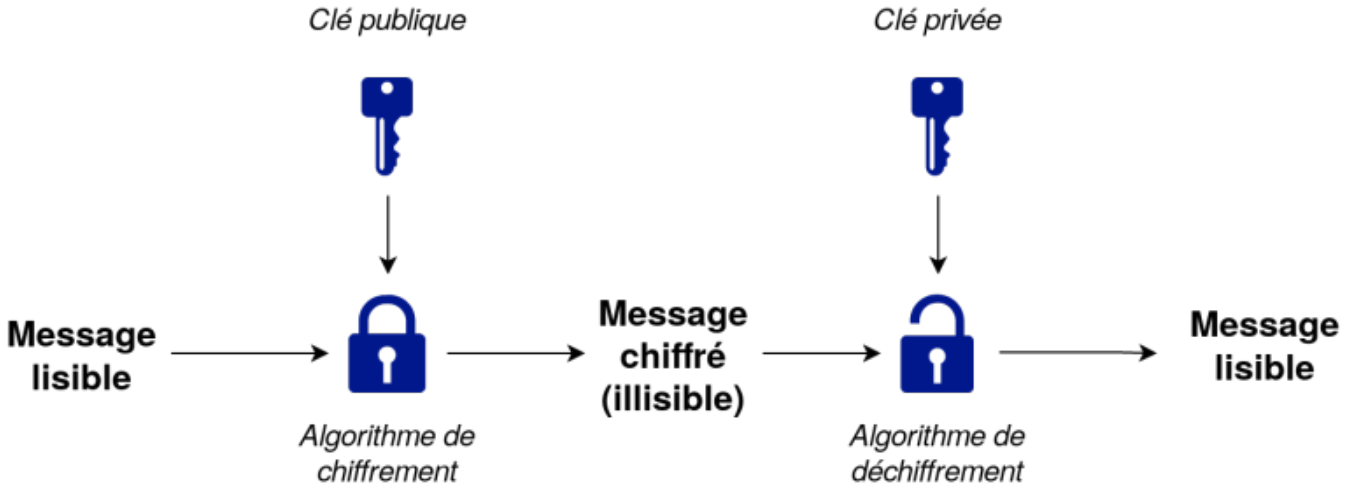
Une paire de clé va être générée avant l'échange de donnée totalisant :

- Une **clé privé** (présente sur le serveur uniquement).
- Une **clé publique** (qui pourra être partagée aux hôtes sans risque).

L'hôte va pouvoir envoyer une donnée de manière confidentielle en chiffrant la donnée à l'aide sa clé publique et de l'algorithme de chiffrement.

Cette donnée chiffrée ne pourra être décodée qu'à l'aide de la clé privé présente sur le serveur et du même algorithme de chiffrement.

Ainsi, pour déchiffrer les échanges dans les deux sens il faudra se munir des deux clés, de l'algorithme de chiffrement et du message chiffrée, ce qui rend la tâche complexe à un pirate pour récupérer la donnée brute initiale.


Clé publique
Clé privée
Message lisible
Message chiffré (illisible)
Message lisible
Algorithme de chiffrement
Algorithme de déchiffrement

[Fondamentaux] ARP / NDP

Introduction

L'**ARP** pour *Address Resolution Protocol* permet la résolution d'adresse MAC à partir des adresses IP, pour que les échanges de paquets puissent s'effectuer entre deux machines d'un même réseau local. Il agit entre la couche 2 et 3 du modèle OSI



Fonctionnement

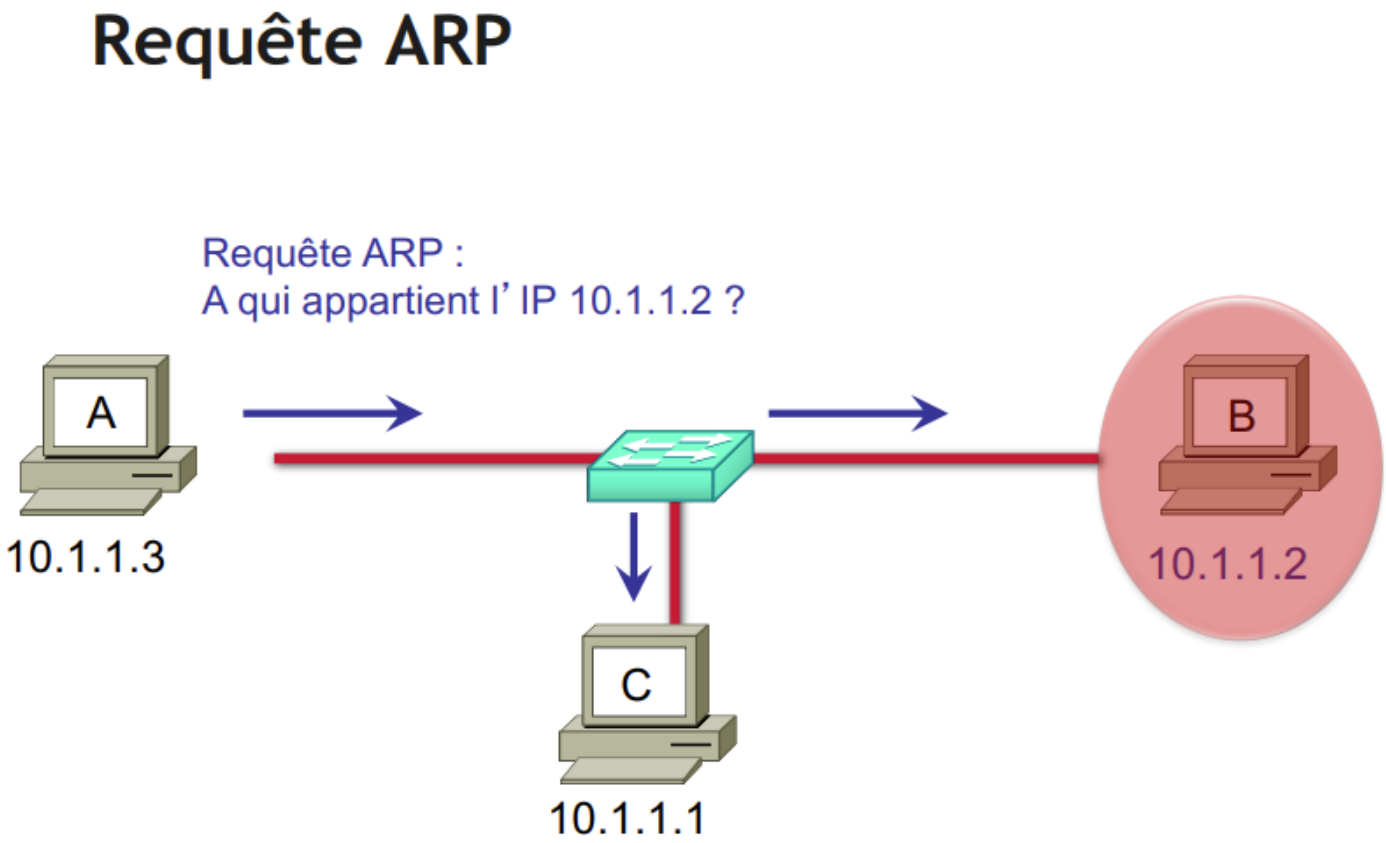
Lors de l'arrivée d'un appareil sur le réseau, celui-ci ne peut pas communiquer avec en utilisant des paquets IP avec les autres machines du réseau car il a besoin de l'adresse MAC de destination pour pouvoir joindre l'hôte de destination.

C'est ici qu'entre en jeu ARP qui va permettre aux hôtes du réseau de remplir leur table ARP qui est une base de donnée locale faisant la **correspondance entre les adresses MAC et les adresses IP.**

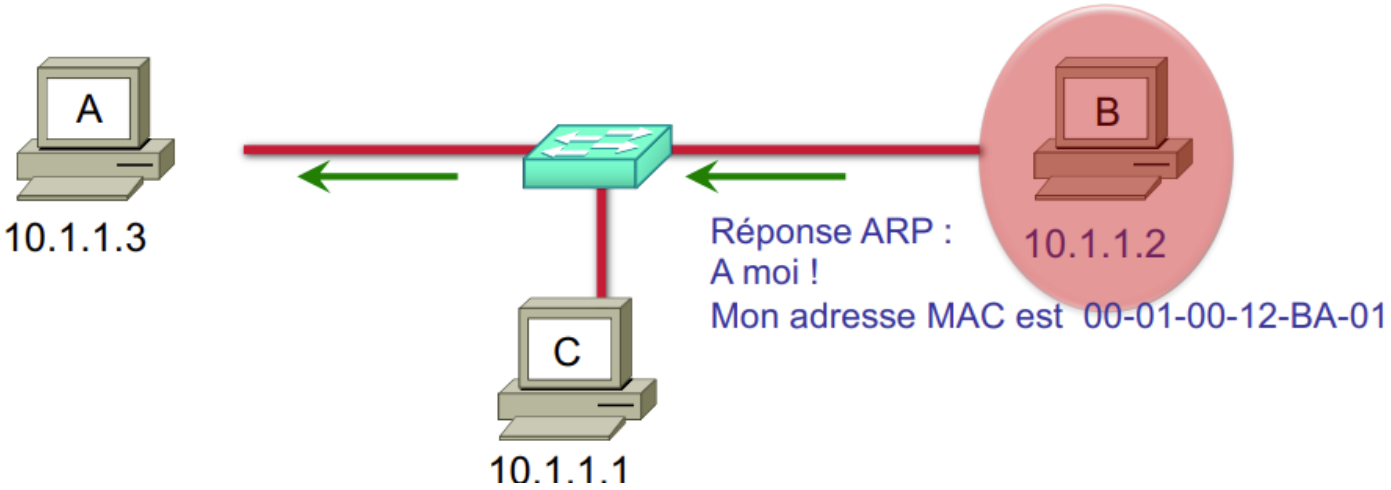
Lorsque l'hôte enverra un paquet à un appareil du réseau dont l'adresse MAC lui est encore inconnue, il va chercher à résoudre l'adresse IP pour trouver la MAC de destination et va devoir effectué un **broadcast de couche 2**, aussi appelé broadcast MAC.

Il va donc envoyer un paquet contenant **12 caractères F** pour que toutes les machines du réseau le reçoive, mais que **seul l'hôte recherché lui réponde** en indiquant son adresse MAC.

Ainsi, l'hôte source de la requête pourra envoyer son paquet (puisqu'il connait désormais la MAC de destination) et ajouter une entrée dans sa table ARP.



Réponse ARP


A
10.1.1.3
B
10.1.1.2
C
10.1.1.1
Réponse ARP :
A moi !
Mon adresse MAC est 00-01-00-12-BA-01

[Fondamentaux] IPv6

Introduction

Cette page explique les changements apportés par **IPv6** par rapport à IPv4 ainsi que son fonctionnement.



Les grands changements

IPv6 règle le problème du manque d'adresses IPv4 puisque ce n'est pas moins de **2^{128} adresses** totales qui sont désormais disponibles.

Il y a tellement d'adresses qu'on s'est permis d'attribuer une adresse "publique" à chaque équipement, ce qui signifie un gain de performance sur les réseaux car il n'y aura **plus besoin du NAT** pour passer d'IP publique à privé.

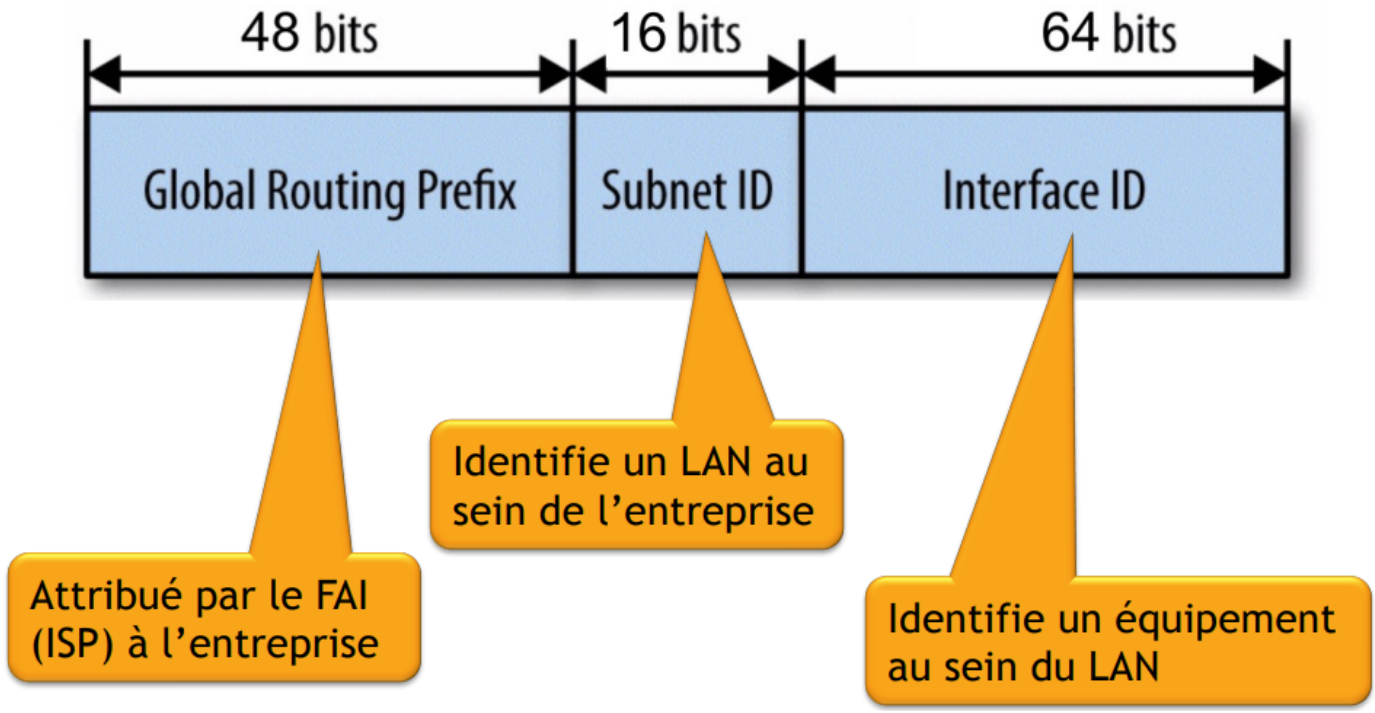
D'ailleurs, **plusieurs adresses** IPv6 peuvent être attribuées à une même interface.

Elle est codée sur **8 blocs de 2 octets écrits en hexadécimal séparés par des ":"** .

IPv6 apporte aussi le support natif du protocole **IPsec**, la **suppression du broadcast** ainsi qu'un **simplification de l'entête**.

Composition d'une adresse

Comme en IPv4, les IPv6 discernent la partie réseau de la partie hôte mais discernent aussi la **partie sous-réseau** :



Les types d'adresses

En IPv6, on distingue 3 types d'adresse IP, le loopback ainsi que l'adresse indéfinie.

Lien local

Certainement le type d'adresse le moins intéressant, il permet de faire communiquer deux équipements sur un **même lien**, une même connexion. Comme le lien local est propre à son segment, une même adresse IP peut être définie sur deux interfaces différentes d'un même appareil.

Ce type d'adresse est **automatiquement attribuée** lors de l'activation d'IPv6 sur un appareil.

Plage d'adresse
FE80::/10

Unique local address

Cette adresse est utilisée au sein d'un réseau ou d'un groupe de réseau d'une organisation.

Plage d'adresse
FC00::0/7

Publique

Il s'agit de l'équivalent de l'IP publique en IPv4. C'est pourquoi elle est valable partout dans le monde.

Plage d'adresse
2000:: /3

Loopback

Il existe aussi en IPv6, seulement sa forme a évoluée :

Adresse
::1

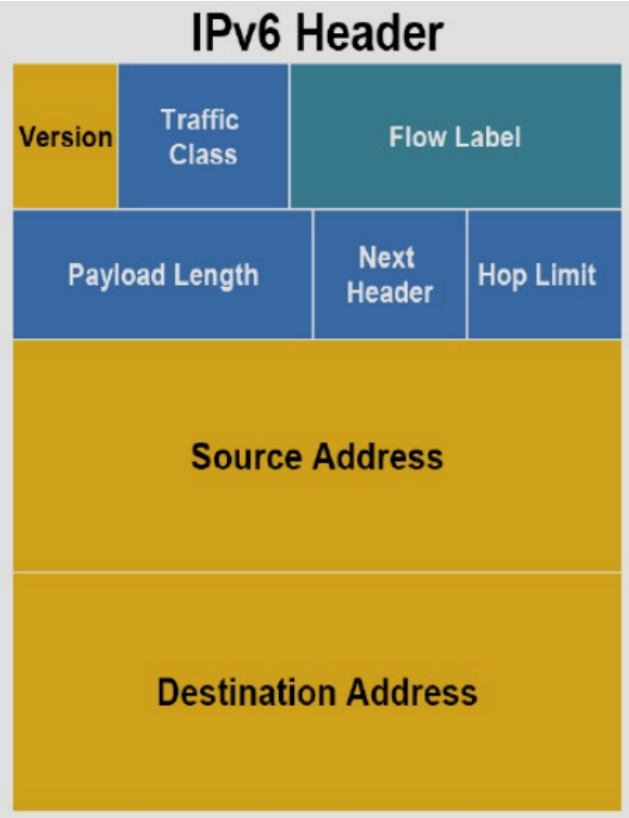
Adresse indéfinie

Souvent utilisée pour définir la gateway, cette adresse sert à désigner l'ensemble des adresses possibles :

Adresse
::

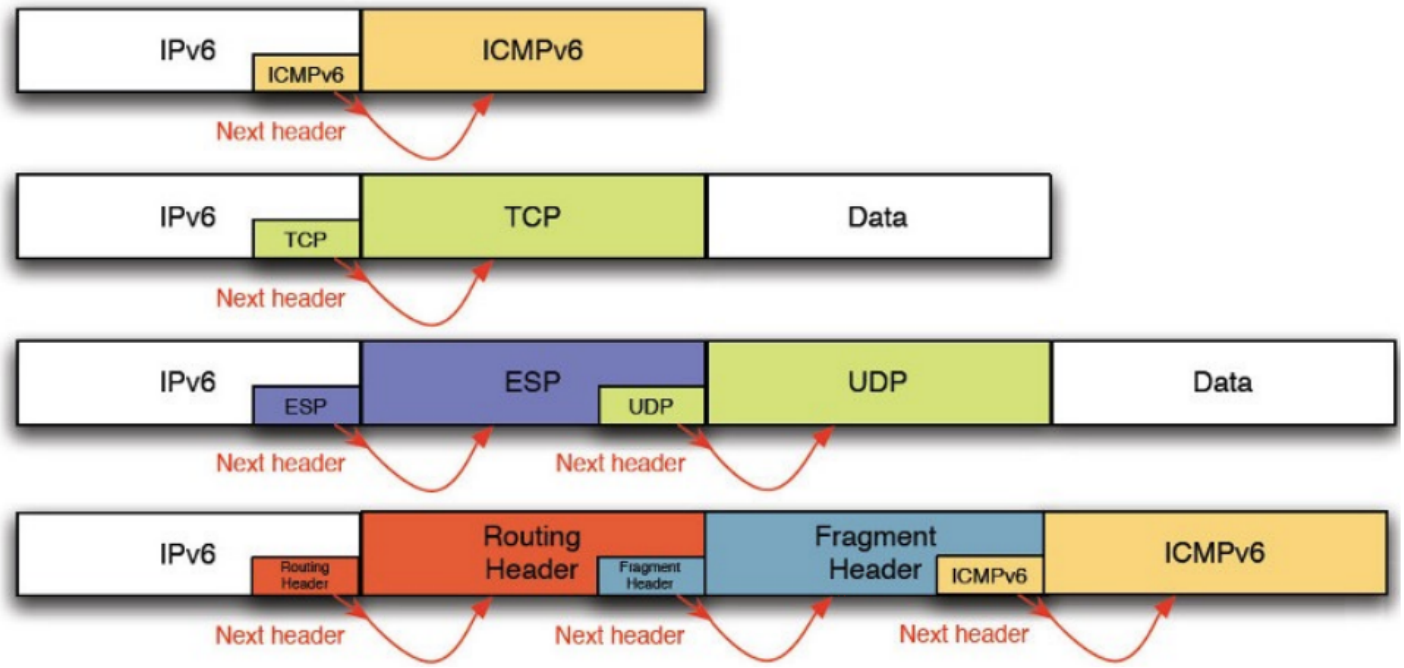
Composition

Voici à quoi ressemble le format d'un paquet IPv6 :



Next header

Le champs *Protocol* de l'IPv4 a été remplacé par la notion de **Next Header** en IPv6, ce qui permet de ne pas limiter les protocoles et ainsi de pouvoir les imbriquer les uns dans les autres :



Remarque : L'extension **ESP** permet de chiffrer la donnée.

Règles de simplification

Une adresse IPv6 peut être simplifiée sous certaines conditions :

- Toute suite de **0000** peut être remplacée par **::**
- Ce remplacement ne peut être effectué qu'**une seule fois**.
- Les 0 en début de section peuvent être supprimés.
- Les 0 en fin de section ne peuvent pas être supprimés.

NDP

En IPv6, ce n'est pas ARP qui se charge de la résolution mais **NDP** pour *Neighbor Disocvery Protocol* qui a un mécanisme semblable.

Une différence majeure se fait notamment sur l'utilisation du **multicast** plutôt que du broadcast (qui n'existe pas en IPv6).

De plus, on ne parle plus de requête ARP mais de **NS** (*Neighbor Solicitation*) ni de réponse ARP mais de **NA** (*Neighbor Advertisement*).

En outre, une vérification est faite pour éviter le problème d'ARP spoofing grâce au **DAD** (*Duplicate Address Detection*) qui va faire un NS avec sa propre adresse MAC pour vérifier que personne n'essaye d'usurper son identité.