ARCHITECTURE MATERIELLE

NSI | 03

RESEAUX

[Marion SZPIEG]

Savoir ce qu'est un réseau ainsi qu'un protocole.

Savoir ce qu'est le modèle TCP/IP (connaître par cœur les 4 couches ainsi que leur rôle)

Connaître la différence de fonctionnement entre un réseau local (LAN) et un réseau étendu (WAN)

Connaître la notion d'encapsulation lors d'une requête entre un client et un serveur

Savoir ce qu'est la commutation de paquets lorsque plusieurs messages arrivent à un répartiteur sur un réseau

Connaître (rôle et couche dans laquelle ils sont) les protocoles suivants : DHCP, DNS, TCP, UDP, IP

1. Qu’est-ce qu’un réseau (network) ?

Définition : un réseau est un ensemble d'ordinateurs et de connexions qui permettent à chaque ordinateur de communiquer avec tous les autres, éventuellement en passant par des intermédiaires.

1.1. Liaisons physiques d’un réseau

Il est possible de faire communiquer deux ordinateurs en les reliant par un simple câble (en général, câble Ethernet avec prise RJ45). On dit alors que ces deux ordinateurs sont en réseau.

Un ordinateur relié à un réseau doit posséder une carte réseau. On reconnaît cette carte réseau de type Ethernet grâce à la prise RJ45 femelle située souvent à l'arrière de l'ordinateur.

Relier 2 ordinateurs peut avoir un intérêt, mais dans la plupart des cas, un réseau sera constitué d'un plus grand nombre d'ordinateurs. Dans ce cas, il est nécessaire d'utiliser un commutateur réseau, souvent appelé switch (même en français). Un switch est constitué de plusieurs prises RJ45. Chaque ordinateur doit être relié au switch par l'intermédiaire d'un câble Ethernet. On peut relier plusieurs ordinateurs sur un switch, et s’il n’y a pas assez de place, on peut relier 2 switchs ensemble pour augmenter le nombre d’ordinateurs en réseau.

Il est aussi possible de mettre plusieurs machines en réseau grâce à des technologies sans fil, par exemple, le wifi. Chaque ordinateur appartenant au réseau sans fil devra posséder une carte réseau wifi. Il sera nécessaire d'utiliser un concentrateur wifi (équivalent du switch en filaire) si l'on désire mettre en réseau plus de deux ordinateurs.

1.2. Établissement d’un dialogue commun : le protocole

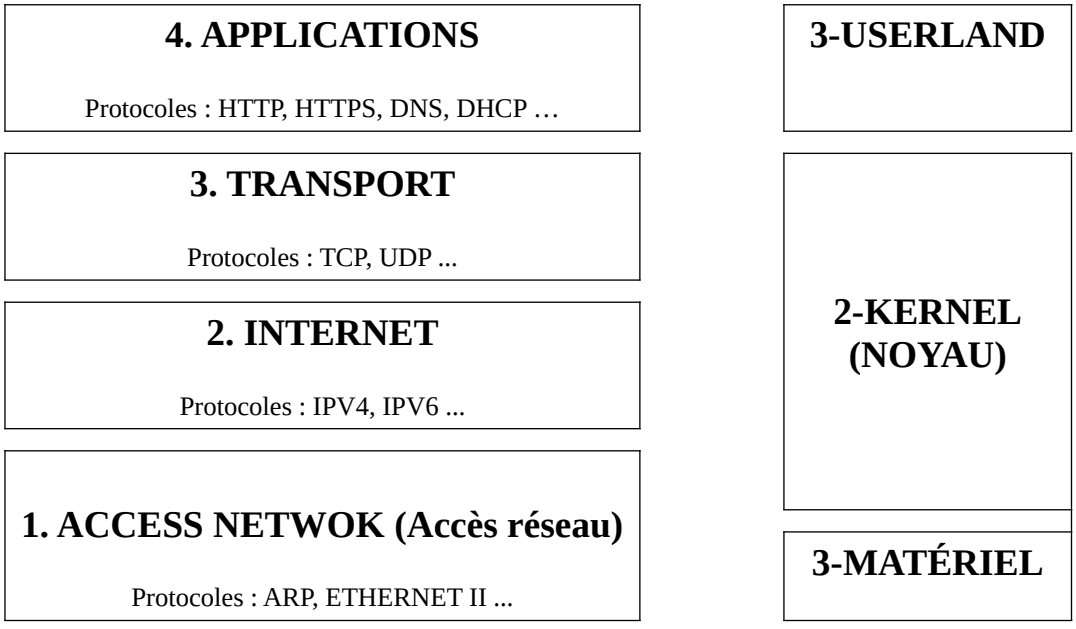
Un protocole est un ensemble de règles qui régit la transmission d'informations dans un réseau. L’ensemble des protocoles est normalisé par l’IETF (Internet Engineering Task Force, organisme international) afin de faciliter la communication dans un réseau entre machines hétérogènes.

La connexion réseau entre 2 machines sur internet est généralement de type client/serveur : l’une des machines envoie une requête (elle est alors machine cliente) et l’autre machine qui répond à la demande est appelée serveur.

2. Le modèle TCP/IP

A l’image de la représentation d’un SE en 3 couches, il existe des modélisations en couche pour représenter le fonctionnement de la partie réseau d’un ordinateur. Les deux plus connus sont le modèle OSI (qu’on n’étudiera pas dans ce cours) et le modèle dit TCP/IP que nous allons étudier cette année.

Ce modèle est composé de 4 couches :



2.1. Couche 1 : ACCESS NETWOK (Accès réseau)

Cette couche permet à un ordinateur A de prendre contact avec un ordinateur B sur le même réseau local ou LAN (Local Area Network). Un réseau local est défini par la méthode utilisée pour qu'un ordinateur A trouve l'adresse d'un ordinateur B sur ce même réseau. Cette méthode sur un réseau local utilise la diffusion (broadcast) : A envoie le message « Je cherche B » sur tout le réseau LAN ; donc toutes les machines du réseau reçoivent ce message. Seule la machine B répond en donnant son adresse. Le protocole qui effectue cette opération est appelé **ARP** (Address Resolution Protocole).

Quelle est cette adresse, et où se trouve-t-elle ?

Il s'agit de l'**adresse MAC** (Media Access Control). Cette adresse est unique et est gravée dans chaque **carte réseau**. Une adresse MAC est composée de 6 octets que l'on exprime sous forme hexadécimale.

Exemple : 30:65:ec:91:79:3a (sous linux) et 30-65-ec-91-79-3a (sous Windows).

Remarque : l'adresse MAC est parfois appelée adresse Ethernet, adresse Hardware ou Adresse matériel

```
marion@marion-Inspiron-7501:~$ifconfig
enp5s0    Link encap:Ethernet  HWaddr e0:69:95:c0:10:59
          inet adr:192.168.1.100  Bcast:192.168.1.255  Masque:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
```

Adresse MAC, matérielle, Hardware (HWaddr)

2.2. Couche 2 : INTERNET

Cette couche permet à un ordinateur A de communiquer avec un ordinateur B ne se trouvant pas sur le même réseau en envoyant un message qui passe de réseau en réseau jusqu'à destination. Il faut donc que A connaisse une adresse de B que l'on peut géolocaliser sur la planète.

Cette adresse est appelée **adresse IP**. Elle dépend du réseau dans lequel la machine se trouve et est contenue dans l'une des cartes réseaux de la machine. Une adresse IP (V4) est composée de 4 octets exprimés en décimal (on parle de notation pointée).

Exemple : 83.166.138.20 (correspond à la machine lfcl-lisbonne.eu hébergeant le site internet du lycée)

```
marion@marion-Inspiron-7501:~$ifconfig
enp5s0    Link encap:Ethernet  HWaddr e0:69:95:c0:10:59
          inet adr:192.168.1.100  Bcast:192.168.1.255  Masque:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
```

Adresse IP, INTERNET (inet)

Cette couche réalise une 2e fonctionnalité très importante : elle découpe les paquets réseaux en plus petits morceaux appelés fragments si le message à envoyer est trop grand. Sur chaque LAN, il existe une taille de paquet maximale appelée MTU (Maximum Transmission Unit). Si un paquet dépasse cette taille, alors le message est envoyé en plusieurs fragments. Sur un réseau LAN avec wifi et/ou connexion filaire, le MTU est par défaut à 1500 octets.

Dans ce cas là, lorsqu'un message dépasse 1500 octets est envoyé, il est découpé en fragments (de 1500 octets maximum). Chaque fragment est envoyé ensuite sur le réseau avec l'information de sa position (offset) dans le message original. Ainsi, la machine destinataire qui récupérera tous les paquets pourra les « recoller » dans le bon ordre (défragmenter). Ce procédé s'appelle la **fragmentation de paquets**.

2.3. Couche 3 : TRANSPORT

Une machine serveur peut offrir plusieurs services aux clients : par exemple, imaginons qu'un processus du serveur du lycée (lfcl-lisbonne.eu) s'occupe du site web et qu'un autre processus du même serveur s'occupe de gérer les boîtes mail. Lorsqu'un message arrive jusqu'à la couche 3 du serveur, le noyau doit décider à quel processus il faut confier ce message. Pour cela, chaque processus serveur attend sur un numéro de port différent (par exemple, le processus web attend sur le port 443, et le processus mail sur le port 993). Donc un message arrivant à la couche 3 contient le numéro du port auquel il est destiné, ce qui permet au noyau de délivrer le message au bon processus.

Pour chaque service rendu par un serveur donné, le numéro de port est normalisé par un organisme international de l'ICANN (Internet Corporation for Assigned Names and Numbers).

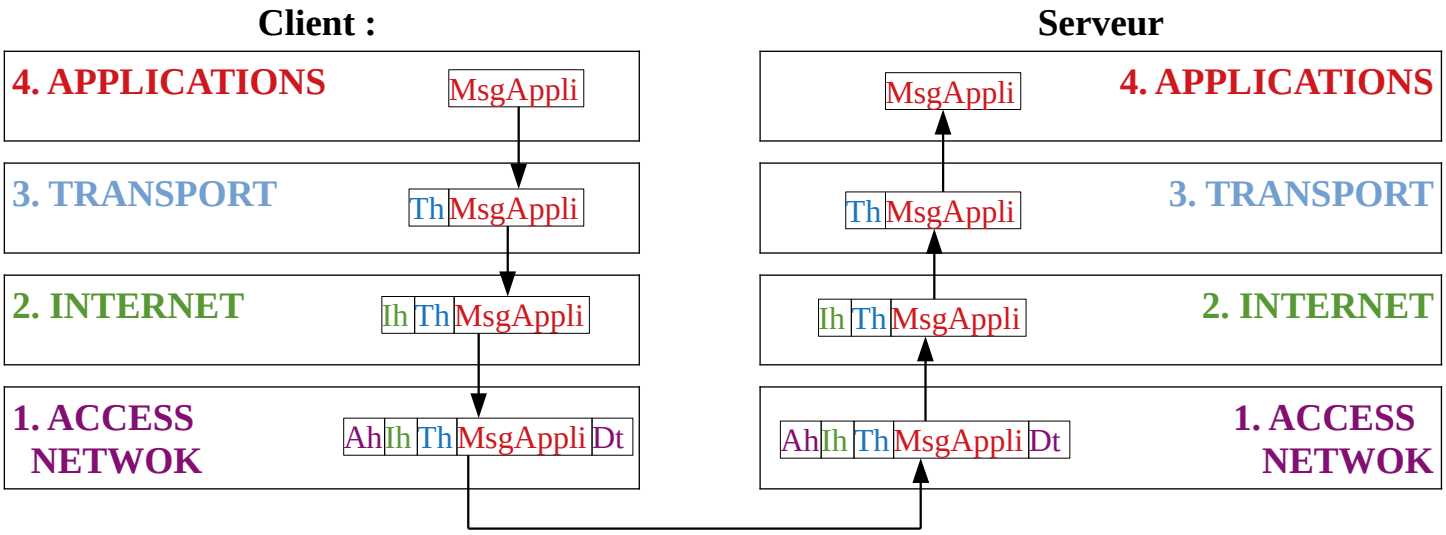
2.4. Couche 4 : APPLICATIONS

Cette couche contient les programmes, aussi appelés applications, qui utilisent des connexions internet (navigateur web (port 433), Steam, ssh (port 22), etc...)

3. Communication dans un réseau

3.1. Réseau local (LAN)

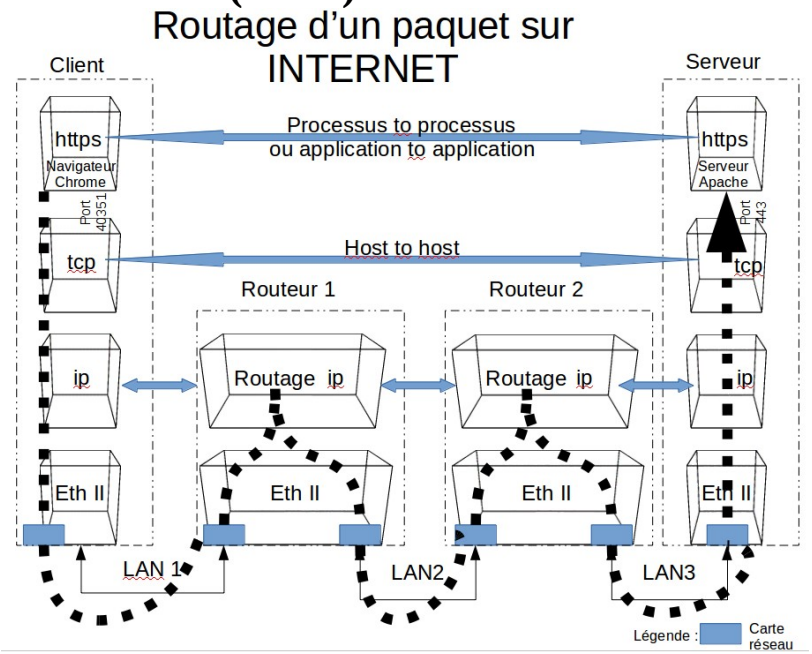
Schéma d'une requête d'une 1ère machine (client) vers une 2e machine (serveur) toutes les deux dans un même réseau local. La 1ère machine cliente connaît préalablement les adresses MAC et IP de la machine serveur en ayant utilisé au préalable le protocole ARP.



Th : Transport header (entête de la couche transport) : contient les numéros de port
Ih : Internet header (entête de la couche Internet) : contient les adresses IP
Ah : Access network headear (…) : contient les adresses MAC
Dt : Delimiteur de trame : ajouté par la couche 1. Permet un contrôle de l'intégrité du message à l'arrivée dans la couche 1 du serveur.

Remarque : un message qui part de la couche 4 du client voit sa taille augmenter par ajout d'information au fur et à mesure qu'il traverse les couches. On parle **d'encapsulation**.

3.2. Réseau étendu (WAN)



Si deux machines qui ne sont pas sur le même réseau local veulent communiquer, la méthode du broadcast ARP ne peut pas fonctionner. Il a donc fallu trouver une autre solution : elle consiste à émettre des paquets en mode « point à point », c'est-à-dire que les paquets vont être confiés à un « routeur » qui lui-même va les transmettre à un autre routeur jusqu'à ce que le message parvienne à la machine destination voulue. Les routeurs contiennent des « tables de routage » qui leur permettent d'envoyer les messages dans la bonne direction. Les adresses IP sont géolocalisées (mais on ne rentrera pas dans les détails).

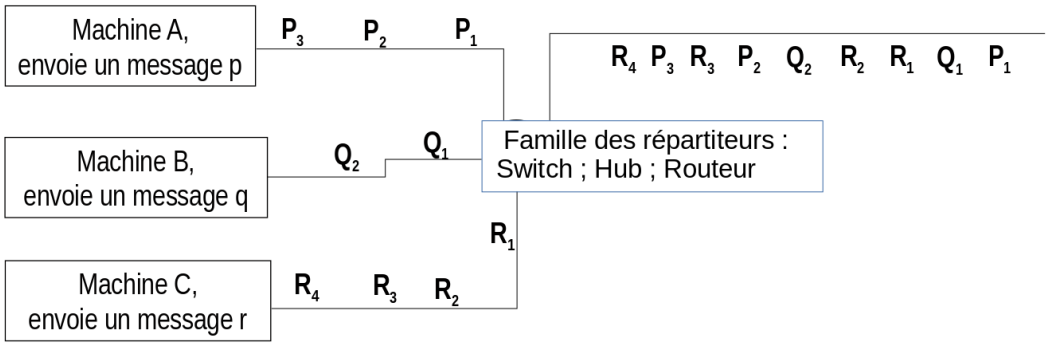
En premier lieu, la machine cliente doit envoyer son message à une machine de son réseau local qui possède une deuxième carte réseau menant vers un réseau extérieur. Cette machine est appelée « passerelle ». Chez vous, la passerelle est la box internet : elle possède une carte réseau qui lui permet d'accéder au réseau de votre fournisseur internet. Chacun des routeurs traversés lit l'entête Internet « header » de manière à réexpédier le paquet dans la bonne direction.

Remarque : il y a en général beaucoup de chemin possibles pour aller d'une machine A à une machine B sur un réseau étendu. Le chemin emprunté par une requête à l'instant t ne sera pas forcément le même que celui emprunté 1 minute ou 1 heure plus tard. Les routeurs sont capable de détecter si certains chemins sont saturés (voir coupés !), et choisir un chemin différent qui fera gagner du temps.

3.3. Commutation de paquets

Sur un réseau, lorsque différents messages arrivent à un même répartiteur (par exemple routeur) et vont emprunter la même voie, le répartiteur pourrait très bien envoyer tous les paquets de la machine A d'abord, puis tous ceux de B puis tous ceux de C. Cela pose un gros problème : si la machine A a beaucoup de paquets à envoyer, les autres attendent… et l'humain n'aime pas attendre !

Pour éviter ce problème, le répartiteur envoie les paquets de A, B et C intercalés. Ce processus est appelé la **commutation de paquets**.



4. Protocoles à connaître

4.1. Protocole DHCP (Dynamic Host Configuration Protocol) (4)

Lorsqu'un ordinateur démarre, dans sa carte réseau, il possède une adresse MAC. Par contre, pour pouvoir dialoguer sur internet, il lui faut une adresse IP qu'il n'a pas par défaut. Il va donc émettre un message en diffusion (broadcast) sur le réseau local afin qu'un serveur lui fournisse une adresse IP. Le protocole utilisé pour faire la demande s'appelle DHCP ; le serveur qui va répondre à cette demande s'appelle par extension un serveur DHCP.

En plus de fournir une adresse IP à la machine, le serveur DHCP fournit 2 autres informations essentielles :

- l'adresse IP de la passerelle
- l'adresse IP d'un serveur DNS (voir partie suivante)

4.2. DNS (Domain Name Systeme) (couche 4)

Les adresses IP étant difficilement mémorisables pour un être humain, on a décidé de donner des noms aux machines. Pour cela, on a divisé les réseaux sur internet en domaines. Un domaine est composé d'un réseau de plusieurs machines administré par la même entité.

Par exemple, les domaines fredpeuriere.com ou szpieg.fr dans lesquels il existe les serveurs respectifs www et marion. Un nom entièrement qualifié d'un serveur sur internet est composé de son nom suivi du nom de domaine auquel il appartient, séparés par un point. Ce qui donne les serveurs www.fredpeuriere.com et marion.szpieg.fr.

Les serveurs DNS de toute la planète sont interconnectés entre eux, et leur travail principal est de renvoyer l'adresse IP qui correspond au nom entièrement qualifié de la machine qui leur a été envoyé. Une commande du shell permet d'interroger un serveur DNS : il s'agit de la commande nslookup.

```
marion@marion-Inspiron-7501:~$ nslookup www.fredpeuriere.com
Server:         127.0.0.53
Address:        127.0.0.53#53

Non-authoritative answer:
www.fredpeuriere.com    canonical name = fredpeuriere.com.
Name:   fredpeuriere.com
Address: 213.186.33.3
```

Les requêtes DNS utilisent le protocole UDP de la couche transport (3) : voir paragraphe suivant.

4.3. TCP (Transport Control Protocol) et UDP (User Datagram Protocol) (couche 3)

Les protocoles TCP et UDP sont des protocoles de la couche transport (3).

Rappel : le rôle principal des protocoles de la couche transport est de gérer les numéros de port afin d'identifier les applications (processus).

Lors d'une connexion client-serveur, l'application serveur se trouve dans un ordinateur identifié sur internet grâce à son adresse IP. Cette application est elle même identifiée dans le serveur par un numéro de port. Par exemple : application serveur web de M. Peurière adresse IP :213.186.33.87 et numéro de port : 443.

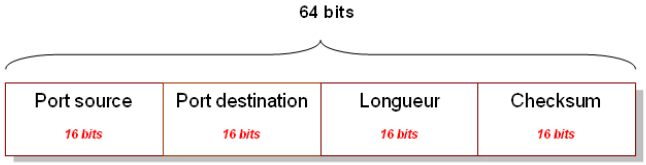
Pour trouver une application sur internet, il faut le couple adresse IP et numéro de port. Ce couple s'appelle une socket (une « prise ») et se note 213.186.33.87:443 . Pour un serveur, le numéro de port est normalisé par l'organisation internationale ICANN (compris entre 1 et 1024). Le port 443 de l'exemple précédent est réservé pour tous les serveurs web communicant avec un protocole sécurisé https.

L'application cliente doit aussi être identifiée de manière unique sur internet pour que le serveur puisse lui répondre. L'application cliente est aussi identifiée par une socket : l'adresse IP de la machine dans laquelle elle se trouve et un numéro de port client choisi aléatoirement par le noyau (hors numéros réservés pour les serveurs).

Exemple : 192.168.1.12:40351.

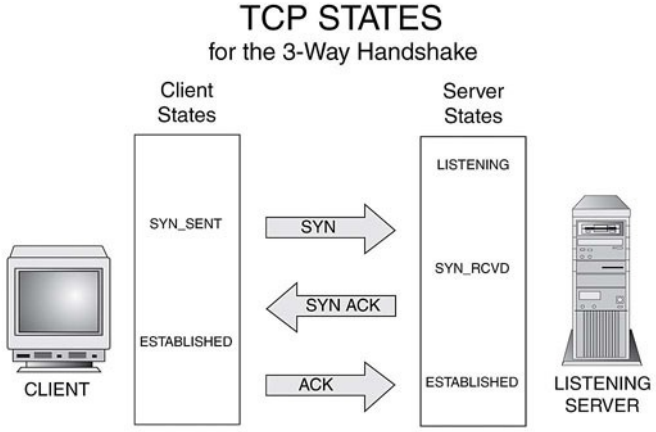
Conclusion : une connexion est composée de 2 sockets : une cliente et une serveur.

Le **protocole UDP** est un protocole minimaliste qui utilise le mode de communication non connecté entre applications. Un mode est dit non connecté si l'application cliente envoie sa demande vers l'application serveur sans vérifier au préalable que celle-ci est en état de répondre. Les paquets envoyés en mode non connecté sont appelés des datagrammes. Un entête UDP contient les informations suivantes :



Le **protocole TCP** est un protocole dit en mode connecté. Le client et le serveur vont échanger 3 messages avant que le client n'envoie sa requête. Ces 3 messages sont appelés les 3 « poignées de main » (three way Handchecks) :

- le client envoie la première poignée de main dans un message appelé « syn » (« je veux me synchroniser avec toi »)
- le serveur répond par une 2e poignée de main appelée « syn-ack » (« j'ai bien reçu ton message (acquittement – acknowledgment) et je veux aussi me synchroniser avec toi »)
- le client envoie la 3e poignée de main appelée « ack » (« j'ai bien compris que tu veux te synchroniser avec moi »)



Puis, le véritable dialogue s'engage. Chaque message échangé sera acquitté par celui qui le reçoit. Ainsi, si un message n'a pas été reçu, l'application émettrice le réexpédiera.

Le protocole UDP est surtout utilisé plutôt pour des échanges courts, contrairement au protocole TCP, mieux adapté à des échanges lourds (au moins plusieurs ko).

4.4. IP (Internet Protocol) (2)

L'entête IP dans les messages échangés entre le client et le serveur contient 3 informations essentielles :

- l'adresse IP de l'émetteur (pour la réponse)
- l'adresse IP du destinataire (pour que les routeurs amènent les paquets sur la bonne machine)
- le cas échéant, si le message original a été découpé en plusieurs fragments, des informations permettant de constituer l'ensemble du message à partir des fragments

5. Schéma complet de l'ouverture d'une connexion client/serveur WEB.