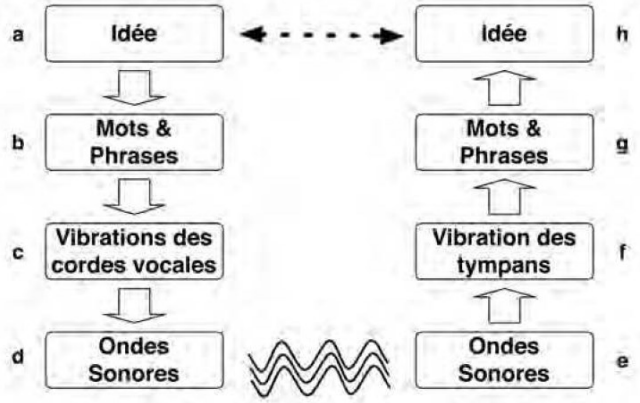
Cours – Introduction à TCP/IP

Objectifs :

- ⇨ Comprendre les concepts de base en matière de réseaux
- ⇨ Savoir comment transite l'information sur un réseau TCP/IP

I – La communication

Nous allons étudier dans ce chapitre comment les ordinateurs s'y prennent pour communiquer entre eux.
Tout d'abord, analysons comment se passe une communication orale entre humain :

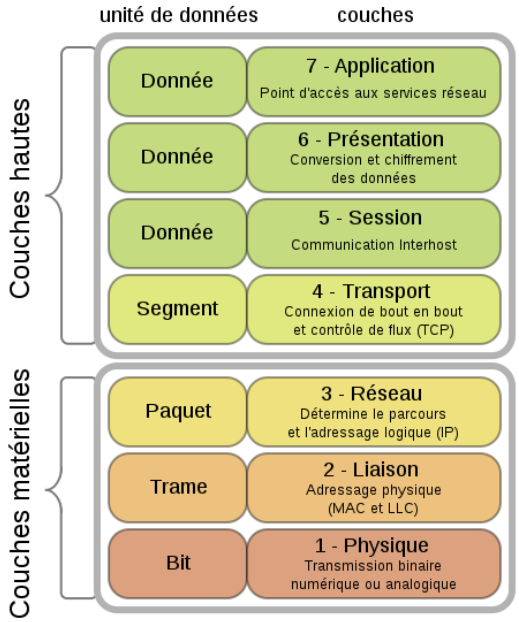


On note qu'il y a un cheminement de l'information qui suit une structure en « couches ». Les modèles pour les ordinateurs reproduisent également cette structure en couches : c'est le modèle OSI.

Ce modèle permet de bien séparer les différents niveaux de la communication afin de les rendre indépendants.

Ainsi la couche application demande juste aux couches inférieures la transmission d'un message sans nécessairement fixer de moyens pour le transmettre et ce sera le rôle des couches inférieures que d'acheminer le message vers le destinataire en s'assurant que le message est bien remis. Suivant le contexte ce sera tel ou tel protocole (ex : TCP, RS232, …), tel ou tel type de liaison (câble éthernet, fibre optique, sans-fil …) qui seront utilisés et cela de manière transparente pour l'application à l'origine de la demande.

L'organisation présentée dans la figure ci-contre n'est pas figée et certains modèles (comme celui de TCP/IP) présentent moins de couches (car ils « fusionnent » certaines couches).



Modèle OSI (source : Wikipédia)

Numéro	Nom	Rôle
Couche 7	Applicative	C'est à ce niveau que sont les logiciels: navigateur, logiciel d'email, FTP, chat...
Couche 6	Présentation	Elle est en charge de la représentation des données (de telle sorte qu'elle soit indépendante du type de microprocesseur ou du système d'exploitation par exemple) et - éventuellement - du chiffrement.
Couche 5	Session	En charge d'établir et maintenir des sessions (c'est à dire débuter le dialogue entre 2 machines: vérifier que l'autre machine est prête à communiquer, s'identifier, etc.)
Couche 4	Transport	En charge de la liaison d'un bout à l'autre. S'occupe de la fragmentation des données en petits paquets et vérifie éventuellement qu'elles ont été transmises correctement.
Couche 3	Réseau	En charge du transport, de l'adressage et du routage des paquets.
Couche 2	Liaison de données	En charge d'encoder (ou moduler) les données pour qu'elles soient transportables par la couche physique, et fournit également la détection d'erreur de transmission et la synchronisation.
Couche 1	Physique	C'est le support de transmissions lui-même: un fil de cuivre, une fibre optique, les ondes hertziennes...

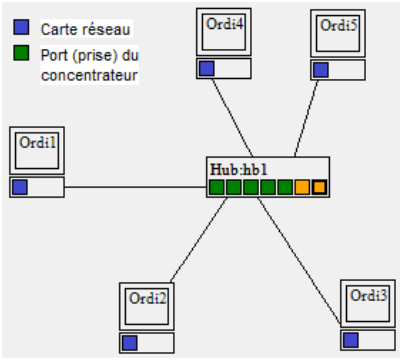
Une explication plus détaillée des couches OSI et la comparaison avec le modèle TCP/IP est disponible sur https://openclassrooms.com/courses/les-reseaux-de-zero/ils-en-tiennent-une-couche-osi-et-tcp-ip-1.

II - Communication en réseau

Comment se passe la communication entre plusieurs ordinateurs (plus de 2) reliés entre eux ? Tout d'abord, il nous faut définir la façon de connecter les ordinateurs entre eux.

1) Interconnexion des ordinateurs

Il existe différentes architectures pour relier de manière filaire des ordinateurs. Nous nous intéresserons ici à la plus fréquente. Elle utilise des sortes de « multiprises réseau » appelés **concentrateurs** (ou hubs en anglais) qui permettent de relier les ordinateurs entre eux.



Chaque ordinateur doit disposer d'une **carte réseau** possédant une prise (ou port) pour y connecter un câble réseau. Ce câble réseau est connecté à l'autre bout à un concentrateur qui va permettre de mettre en relation les différents ordinateurs du réseau.

Chaque ordinateur (ou plutôt chaque carte réseau) a une adresse unique qui permet de l'identifier : c'est l'adresse MAC (Media Access Control).

Application 1 :
Ouvrir une ligne de commande sur l'ordinateur (touche windows+R, puis commande « cmd » ou aller dans le menu démarrer/Accessoires/Invite de commande). Taper la commande `ipconfig /all` et noter l'adresse physique de la première carte réseau (du type D8-CE-7A-BF-66-24). Vérifier avec vos voisins que cette valeur est unique.

Pour comprendre pourquoi il est nécessaire que chaque carte réseau ait sa propre adresse et comment fonctionne un concentrateur, visualiser la présentation adressemac (en vidéo) dans le répertoire de l'activité.

A chaque fois qu'un ordinateur veut communiquer, il envoie une trame sur le réseau, et celle-ci est diffusée à l'ensemble des postes (qui l'ignorent s'ils ne sont pas destinataires). Que se passe-t-il si un poste veut émettre une trame alors qu'un autre est déjà en train d'émettre ?
Dans ce cas, on dit qu'il y a *collision* des trames et ceci rend le message inintelligible. Chaque ordinateur attend alors un délai variable et aléatoire pour tenter de ré-émettre sa trame.
On comprend aisément que plus le nombre de postes du réseau augmente, plus les collisions seront fréquentes. De plus ce type de fonctionnement ne permet pas le dialogue simultané de plusieurs paires de postes.
Pour solutionner partiellement ce problème, on peut utiliser des **commutateurs** (switchs en anglais) plutôt que des concentrateurs.
Voir la présentation limite_des_commutateurs (vidéo dans le répertoire de l'activité) pour plus d'information.

Le commutateur analyse le trafic sur ses ports pour déterminer l'adresse MAC de chaque carte réseau connecté à ses ports. Ainsi lorsqu'une trame arrive qui s'adresse à une certaine adresse MAC, il ne transmet la requête qu'au port concerné et ne la répète pas sur les autres ports. On limite ainsi fortement les risques de collisions et on peut faire plusieurs requêtes en parallèle. Ainsi sur le schéma précédent, Ordi1 et Ordi2 peuvent dialoguer pendant que Ordi3 et Ordi4 échangent entre eux sans que les deux dialogues ne se gênent.

Mais les commutateurs peuvent-ils gérer autant d'ordinateur que l'on veut ? Non, car en regardant la présentation limite_des_commutateurs on a vu que chaque commutateur devait mémoriser les adresses MAC de tous les postes du réseau pour savoir où transmettre l'information, or la mémoire de ces appareils n'étant pas illimitée, on ne peut pas espérer gérer des millions de postes.

Pour résoudre le problème, on va faire appel au protocole IP qui est décrit dans la présentation adresseIP (voir vidéo dans le répertoire de l'activité).

2) Internet

On peut ainsi connecter des réseaux d'ordinateurs entre eux. Cette interconnexion des réseaux, ou réseau de réseaux est ce qu'on appelle l'internet (inter – net : entre les réseaux).

Voyons ensemble comment un message est « routé », c'est-à-dire acheminé entre deux points d'un réseau avec un exemple pratique.
La commande « traceroute » permet de déterminer le chemin emprunté par les paquets IP pour aller du poste local à une adresse internet donnée. Nous allons l'utiliser pour déterminer la route empruntée pour relier notre ordinateur à différents serveurs.

Application 2 :

1) Dans l'invite de commande, exécuter la commande `tracert 217.160.0.126` (adresse du site du lycée : http://www.lyceejoliotcurie77.fr/). Combien d'étapes ou sauts (hop en anglais) sont-elles nécessaires à l'acheminement du message ?

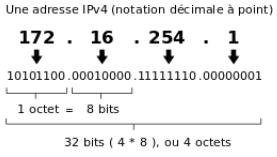
2) Exécuter ensuite la commande `tracert 200.7.161.56` (adresse du site de l'université de La Paz : www.umsa.bo). Par quel autre pays passe la requête ? Combien de saut a t-il fallut ?

Aide : Les noms des routeurs contiennent souvent un code à 3 lettres qui correspond au code de l'aéroport le plus proche géographiquement. Par exemple `ge5-2.mpr2.cdg2.fr.above.net` se situe vers Paris car il contient le code **cdg** qui désigne l'aéroport Charles De Gaule. De même le routeur ayant pour adresse `so-7-0-0.mpr1.ams5.nl.above.net` se situe près d'Amsterdam (code **ams**). Une recherche avec les mots-clé « Aéroport » et le code à 3 lettres ou sur le site https://fr.wikipedia.org/wiki/Liste_des_codes_AITA_des_a%C3%A9roports/A peut vous permettre de retrouver le lieu où se trouve le routeur.
Pour déterminer la localisation (approximative) correspondant à une adresse IP, on peut aussi utiliser des outils en ligne comme GeoData tool.

Il existe aussi des outils graphiques plus visuels comme Open Visual Traceroute mais qui nécessite une installation sur l'ordinateur.

3) Adresse IP

Chaque ordinateur sur le réseau doit donc avoir une adresse IP unique pour pouvoir le situer.
Dans ce document nous n'allons aborder que les adresses de type IPv4. On utilise aussi des adresses IPv6, mais elles ne sont pas encore généralisées auprès du grand public.



L'adresse IP est constituée de 4 octets, présentés sous la forme de 4 nombres de 0 à 255 séparés par des points. Elle suit des règles particulières (voir https://fr.wikipedia.org/wiki/Adresse_IP), dont nous allons voir les principales :

- Les adresses sont organisées en réseaux, sous-réseaux et numéro d'hôte, un peu comme une adresse postale est organisée en ville, rue et numéro dans la rue) :
  Le premier octet de l'adresse est son numéro de réseau, le deuxième celui du sous-réseau et les deux derniers constituent l'adresse du poste dans le sous-réseau.
  Exemple : 154.66.12.57 Réseau : 154
  Sous-réseau : 66
  Numéro de poste : 12.57 = 12*256 + 57 = 3129

- Certaines adresses sont réservées. Ainsi il existe plusieurs classes d'adresses privées. Ce sont des adresses qui correspondent à des réseaux locaux et ne sont donc pas diffusées (routées) sur internet.
  Adresses privées : 10.x.y.z , 192.168.x.y , 172.16.x.y
  Adresse locale : 127.0.0.1 cette adresse appelée « localhost » désigne le poste local

- Dans chaque réseau il existe trois adresses particulières :
  - ⇨ La première adresse (celle qui finit par 0) désigne le réseau lui-même et ne peut donc pas être attribuée à un poste.
  - ⇨ La première adresse possible (celle qui finit par 1) est *généralement* celle du routeur qui permet de relier ce réseau aux autres réseaux.
  - ⇨ La dernière adresse du réseau (qui ne finit que par des 1 en binaire) est l'adresse de multidiffusion (*Broadcast* en anglais) : tout message envoyé à cette adresse sera diffusé à tous les postes du sous-réseau.

Exemple :
Soit le réseau local dont l'adresse commence par 1100 0000 . 1010 1000 . 00xx xxxx . xxxx xxxx en binaire
L'adresse du réseau sera : 1100 0000 . 1010 1000 . 0000 0000. 0000 0000 ou 192.168.0.0 en décimal
Le routeur devrait se trouver à : 1100 0000 . 1010 1000 . 0000 0000. 0000 0001 ou 192.168.0.1 en décimal
Adresse de broadcast sera : 1100 0000 . 1010 1000 . 0011 1111. 1111 1111 ou 192.168.63.255 en décimal

Comment connaitre l'adresse IP de mon poste ?

Application 3 :
Utiliser la commande `ipconfig` de l'invite de commande pour déterminer l'adresse IP de votre poste. Elle est en principe du type 172.16.x.y c'est-à-dire qu'il s'agit d'une adresse privée appartenant au réseau de l'établissement. Vérifiez avec vos voisins qu'elle est bien unique.

Lorsqu'on communique à l'extérieur du réseau (sur internet), l'adresse utilisée est l'adresse externe du routeur. Pour la connaitre, on utilise des sites comme http://ipcheck.com/ qui renvoient cette adresse. Si on veut communiquer entre deux ordinateurs à travers internet, c'est ces adresses « externe » qu'il faut utiliser.

4) Masque de sous-réseau

La commande ipconfig permet également de connaitre le masque de sous-réseau. Ce masque est utilisé pour déterminer si une adresse IP est sur le même sous-réseau (et que l'on peut la joindre directement) ou s'il faut s'adresser au routeur pour atteindre une adresse appartenant à un autre réseau (sur internet).

a. Détermination de la valeur du masque

Le masque est une valeur binaire où on met à 1 tous les bits correspondant au réseau ou au sous-réseau et à 0 tous ceux qui correspondent à l'adresse dans le réseau.

Exemple : On utilise le réseau 192.168.25.x où x peut varier de 0 à 255.
Les octets correspondant au sous-réseau (**net-ID**) sont donc les 3 premiers (192.168.25) et le dernier octet correspond au numéro du poste dans le sous-réseau (**HOST-ID**).
Le masque sera donc 1111 1111 . 1111 1111 . 1111 1111 . 0000 0000

tous les bits à un pour le sous-réseau — tous les bits à 0 pour le numéro de poste

Soit un masque valant 255.255.255.0 en décimal

b. Utilisation du masque

Pour savoir si une adresse à joindre est sur le même sous-réseau, il suffit de faire un ET logique entre l'adresse de notre poste et le masque de sous-réseau, puis entre l'adresse distante et le masque de sous-réseau et de comparer les deux résultats. S'ils sont identiques, c'est que l'adresse est dans le même sous-réseau.

Exemple : Notre poste a pour adresse 192.168.25.23 et cherche à joindre 192.168.25.113 dans le sous-réseau ayant pour masque 255.255.255.0

```
Notre IP :               1100 0000 . 1010 1000 . 0001 1001 . 0001 0111    192.168.25.23
ET masque sous-réseau :  1111 1111 . 1111 1111 . 1111 1111 . 0000 0000    192.168.25.0
                         1100 0000 . 1010 1000 . 0001 1001 . 0000 0000  = 192.168.25.0

IP à contacter :         1100 0000 . 1010 1000 . 0001 1001 . 0111 0001    192.168.25.113
ET masque sous-réseau :  1111 1111 . 1111 1111 . 1111 1111 . 0000 0000    192.168.25.0
                         1100 0000 . 1010 1000 . 0001 1001 . 0000 0000  = 192.168.25.0
```

Le résultat des deux opérations est le même, donc l'IP à contacter est sur le même sous-réseau.

On cherche maintenant à joindre 145.8.77.202 avec le même masque de sous-réseau.

```
Notre IP :               1100 0000 . 1010 1000 . 0001 1001 . 0001 0111    192.168.25.23
ET masque sous-réseau :  1111 1111 . 1111 1111 . 1111 1111 . 0000 0000    192.168.25.0
                         1100 0000 . 1010 1000 . 0001 1001 . 0000 0000  = 192.168.25.0

IP à contacter :         1001 0001 . 0000 1000 . 0100 1101 . 0111 0001    192.168.25.23
ET masque sous-réseau :  1111 1111 . 1111 1111 . 1111 1111 . 0000 0000    145.008.77.202
                         1001 0001 . 0000 1000 . 0100 1101 . 0000 0000  = 145.008.77.0
```

Le résultat des deux opérations est différent, donc l'IP à contacter n'est pas sur le même sous-réseau, il faudra donc faire appel au routeur pour la joindre.

Application 4 :

1) A l'aide de la commande `ipconfig` déterminer le masque de sous-réseau utilisé par votre poste.

2) Quel est alors le nombre maximum de postes que peut contenir ce sous-réseau ?

3) Quelle est l'adresse du routeur auquel transférer les paquets si l'adresse de destination n'est pas dans le sous-réseau (il s'agit de la passerelle par défaut) ?

5) Les ports

Comment faire tourner plusieurs services différents (serveur web, serveur de messagerie, chat, …) sur le même poste (et donc avec la même IP) ? En effet si toutes les requêtent arrivent à la même adresse, comment le serveur pourra-t-il savoir à quel service est destiné le message qu'il vient de recevoir ?
La solution est dans l'utilisation de ports en complément de l'adresse IP. Chaque carte réseau, en plus d'avoir une adresse IP possède 65536 ports numérotés de 0 à 65535 et chaque service peut donc utiliser un (ou plusieurs) ports différents.

Ainsi pour joindre par exemple un serveur web et demander une page html, le navigateur doit adresser une requête au port 80 de l'hôte avec le protocole http. Le gestionnaire du site web, lui, s'il souhaite envoyer au serveur une nouvelle page web utilise le port 21 avec le protocole ftp (**F**ile **T**ransfert **P**rotocol) pour envoyer son fichier.

On peut également spécifier un port particulier dans une URL en indiquant son numéro après le signe '`:`' derrière le nom de domaine. Cela se fait souvent pour accéder au serveur web de certains équipements réseau (box, routeur, nas, …)

Port	Service	Description
21	Port de contrôle FTP	Le FTP est utilisé pour le transfert des fichiers.
22	SSH (**S**ecure **SH**ell)	Utilisé dans certaines applications de connexion à distance
23	Telnet	
25	SMTP	Courrier sortant vers un serveur SMTP (**S**imple **M**ail **T**ransfer **P**rotocol)
53	Domain Name Server (DNS)	Voir plus loin l'explication détaillée
68	DHCP	Utilisé pour une configuration automatique des adresses IP
80	World Wide Web, http	Navigation sur Internet en HTTP (**H**yper**T**ext **T**ransfert **P**rotocol)
110	POP3 (**P**ost **O**ffice **P**rotocol)	Courier entrant
443	HTTPS	Navigation sur certains sites sécurisés

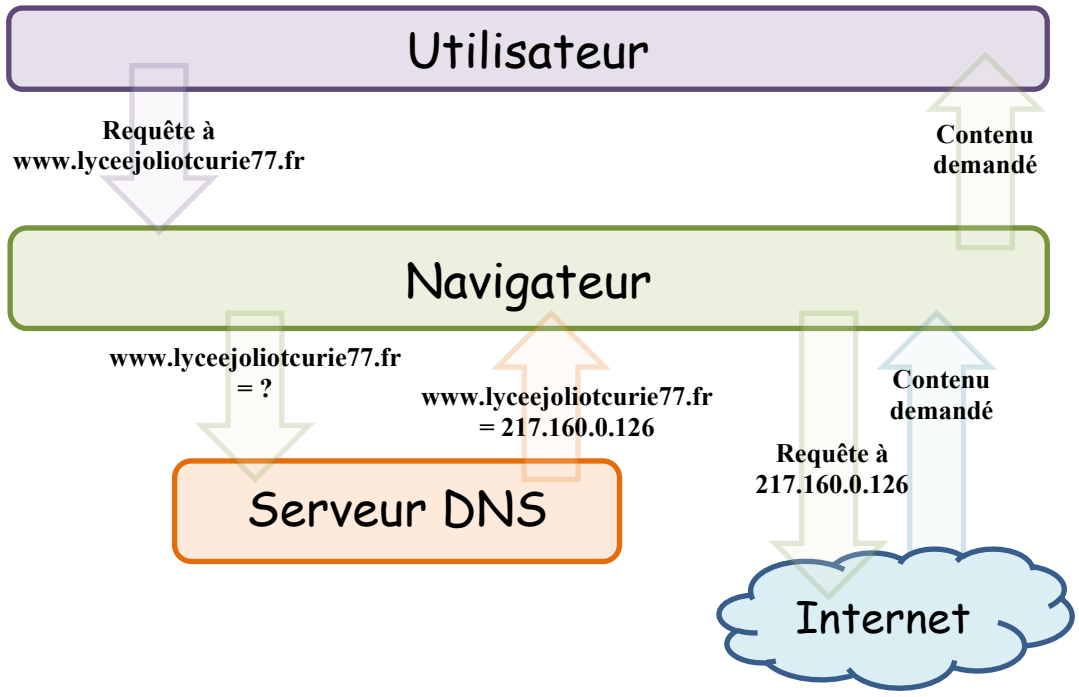
Ex : http://monsite.fr:8080/siteweb/index.html

La partie en bleu est le protocole (ici http, le protocole servant à échanger des pages html)
La partie en vert est le nom de domaine
La partie en rouge est le numéro de port (dans cet exemple c'est le port n°8080)
La partie en violet est le chemin de la page (fichier html) dans l'arborescence du serveur.

Davantage d'information sur http://www.commentcamarche.com/contents/528-port-ports-tcp-ip.

6) DNS

Joindre un serveur sur internet doit se faire avec son adresse IP, mais il peut être difficile de se souvenir par cœur de cette suite de 4 nombres. Il est plus simple de retenir www.lemonde.fr[1] que 93.184.220.239.
C'est là qu'interviennent les DNS (Domain Name Server). Ces serveurs (dont les adresses IP sont connues) vont faire la traduction entre nom de domaine (www.lemonde.fr) et l'adresse IP associée (93.184.220.239).
Ainsi lorsqu'on tape une adresse ou qu'on clique sur un lien, le navigateur va interroger le serveur DNS pour connaitre l'adresse IP correspondante, puis ensuite il va contacter le site grâce à l'adresse IP qu'il vient d'obtenir.



[1] C'est ce qu'on appelle un FQDN (Fully Qualified Domain Name) ou nom de domaine qui qualifie un ou plusieurs ordinateurs répondant à la même adresse IP.

La commande `ping` permet de tester la rapidité de connexion vers une adresse IP en mesurant le temps que met un petit paquet de données à faire l'aller-retour vers cette adresse. On peut l'utiliser pour déterminer l'adresse IP d'un site dont on connait le nom de domaine.

Application 5 :
A l'aide de l'invite de commande, faire un `ping` vers www.amazon.com, www.amazon.fr, www.amazon.co.uk, www.amazon.es, puis vers www.google.com, www.google.fr, www.google.be, www.google.es … Que constatez-vous dans chacun des deux cas ?

7) Who's who

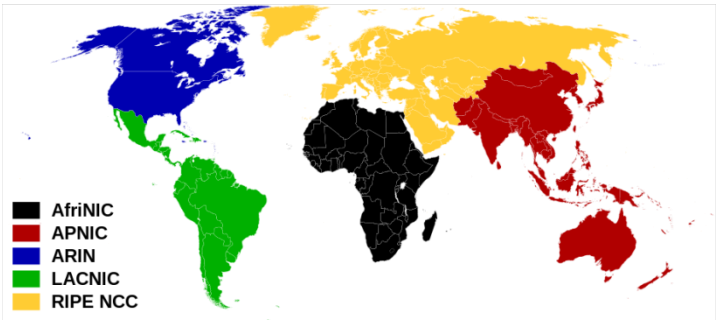
Comment retrouver le nom de domaine quand on connait l'adresse IP ? Tout simplement en adressant une requête de *reverse DNS* au serveur DNS (ou reverse DNS lookup ou encore rDNS). Dans Windows cela se fait avec la commande `nslookup`.

Application 6 :
Dans l'invite de commande, utiliser `nslookup` pour déterminer les noms de domaine des IP suivantes : 198.27.92.1, 212.27.48.10, 195.221.251.178 et 64.91.226.82

NB : La recherche de nslookup se fait avec le serveur DNS par défaut de la connexion. On peut forcer l'utilisation d'un autre serveur DNS en spécifiant son adresse IP juste après l'adresse IP cherchée.
Exemple : `nslookup 217.160.0.126 80.67.169.12` fait une recherche du nom attribué à l'IP 217.160.0.126 en utilisant le serveur DNS de FDN (80.67.169.12).

La recherche DNS inverse amène souvent des résultats déroutants car les grandes infrastructures du web utilisent des mécanismes permettant d'attribuer plusieurs serveurs et plusieurs adresses IP à un même nom de domaine. On tombe aussi sur des reverse DNS « génériques » du type `217-160-0-126.elastic-ssl.ui-r.com` qui sont utilisés par la plupart des FAI (ici `elastic-ssl.ui-r.com` est le serveur rDNS du fournisseur d'accès et l'adresse IP est indiquée à l'envers (`217-160-0-126` au lieu de `126.0.160.217`).

Comment obtenir plus d'informations sur un nom de domaine ou une IP ? Tout simplement en s'adressant à l'autorité qui gère les adresses IP et en faisant une requête *whois*.
Les noms de domaine sont gérés par des autorités régionales (voir carte). Pour l'Europe c'est le RIPE (Réseau IP Européen).



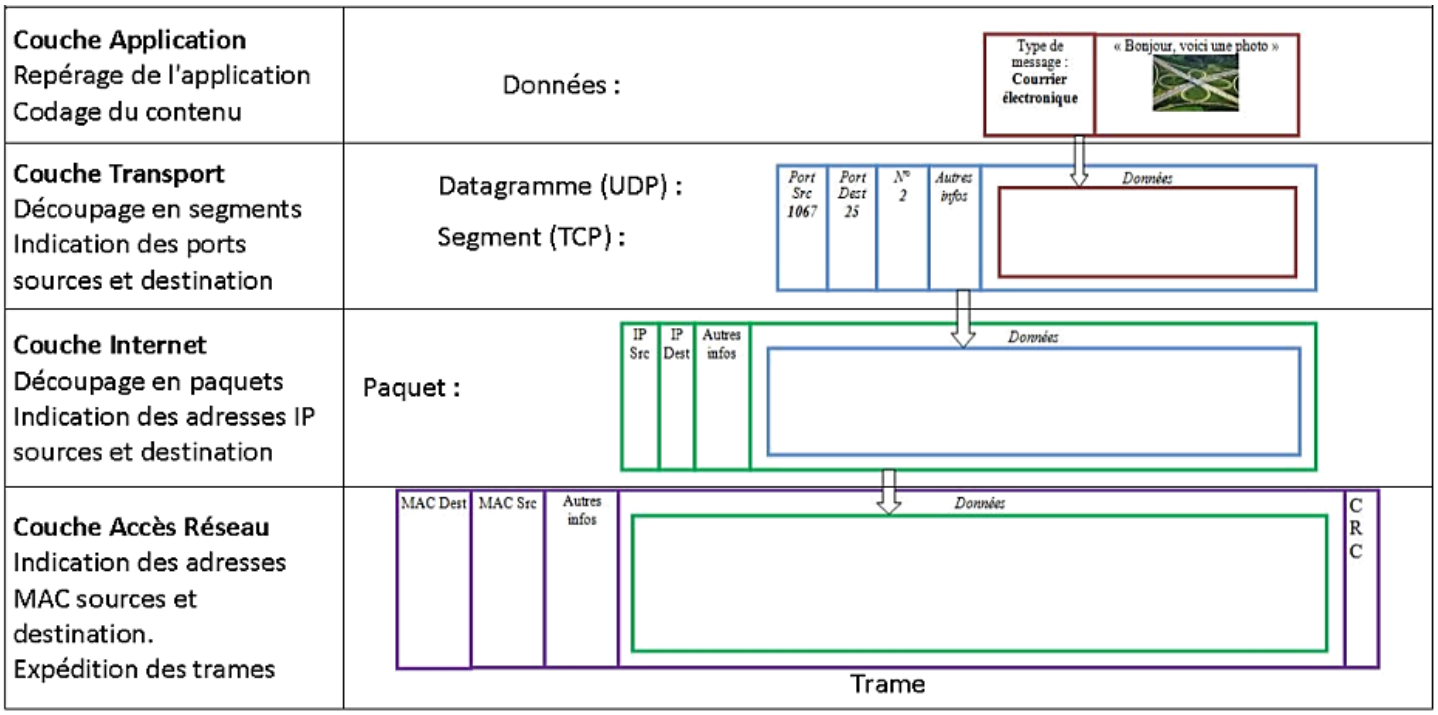
Certains sites proposent également des services de whois simplifiés. Exemple : https://www.whois.com/whois.

Application 7 :
Utiliser les services du site whois.com pour retrouver les informations sur le site du lycée (www.lyceejoliotcurie77.fr).

III – Transmission de l'information à travers un réseau

1) Fragmentation et encapsulation de l'information

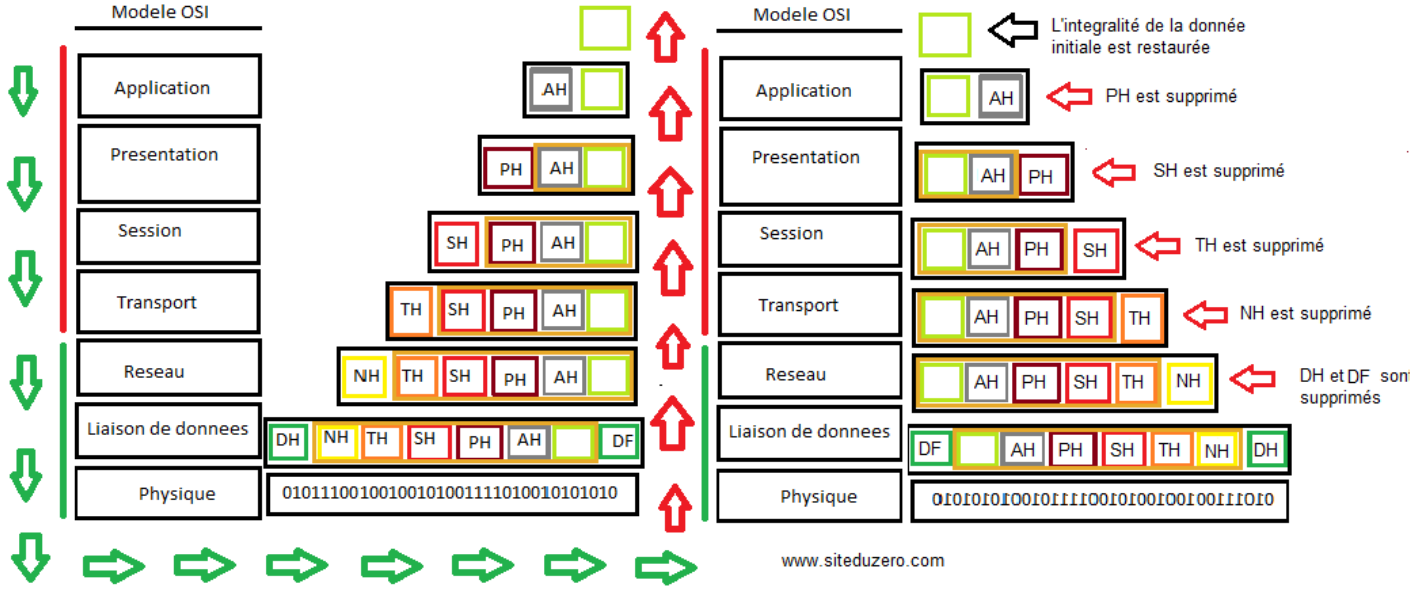
Lorsqu'une application doit transmettre des données (par exemple un document texte) à un autre poste du réseau, elle envoie ses données à la couche inférieure du modèle OSI (couche données) qui va formater les données, éventuellement les chiffrer et les transmettre à la couche inférieure qui va initier une communication vers le poste distant et transmettre les données à la couche inférieure (transport – TCP) qui va se charger de fragmenter les données en plusieurs petits paquets et d'envoyer ces paquets à la couche inférieure (Réseau – IP) qui elle va gérer l'envoi des paquets en spécifiant l'adresse IP du poste à joindre et transmettre ces paquets à la couche inférieure (Liaison) qui elle retrouvera l'adresse MAC de la machine à joindre (si elle est sur le même réseau local) et transférera ses paquets à la couche physique. A chaque étape, les différentes couches vont retransmettre l'information qu'on leur a donné en y rajoutant des informations servant à gérer le flux de données (par exemple la couche TCP rajoute les numéros de port source et destination, la couche IP rajoute l'adresse IP source et destination, la couche liaison rajoute l'adresse MAC source et destination, …). Ce procédé est connu sous le nom d'encapsulation.



http://silanus.fr/sin/formationSTI2D/ModuleReseau/co/Reseau_19.html

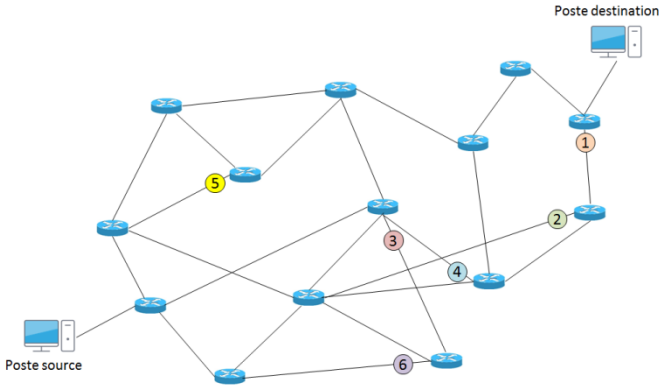
Ces opérations se font en sens inverse au niveau du destinataire.

Mécanisme d'envoi et de réception :



2) Transmission via un réseau de réseau (internet)

Pour transmettre les paquets sur internet, le poste source va s'adresser à un routeur qui va se charger de transférer les paquets de proche en proche jusqu'à la destination. La route exacte prise par les paquets peut changer, y compris d'un paquet à l'autre, car les routeurs utilisent des algorithmes complexes pour déterminer à quel routeur ils doivent confier le paquet à transmettre. Cela dépend notamment du temps mis par le paquet pour aller d'un routeur à un autre, du débit de chaque liaison et de son encombrement.



Visualisez un exemple de routage avec le diaporama « Exemple de routage.ppsx » disponible dans le répertoire de l'activité.

3) Simulation d'un réseau

On va simuler le fonctionnement d'un réseau avec le logiciel Filius.

Application 8 :
Faire le TP décrit dans la page https://pixees.fr/informatiquelycee/n_site/snt_internet_sim1.html