

9.4

Le modèle TCP/IP (4 couches)

NSI 1ÈRE - JB DUTHOIT

TCP/IP est né de la réflexion de chercheurs américains suite à un problème posé par l'armée américaine.

Le problème posé par l'armée américaine

L'armée américaine dispose de plusieurs bases sur le territoire. Chacune de ces bases possède des centres reliés entre eux par des réseaux locaux différents. Cependant ces centres informatiques doivent échanger des informations. Les bases sont reliées les unes aux autres par des câbles.

La question était de trouver un moyen pour que l'information puisse circuler entre ces bases même si certains des chemins empruntables étaient détruits. Il fallait donc trouver un système permettant de retrouver des chemins (routes) qui se reconfiguraient automatiquement en cas de coupure des liaisons.

De cette recherche est née IP (Internet protocol). IP est un protocole qui permet d'envoyer des informations élémentaires d'une machine à une autre.

Cependant, l'information ne part pas de la machine, mais d'une application fonctionnant sur la machine. Pour résoudre ce problème, on a développé un autre protocole : le TCP.

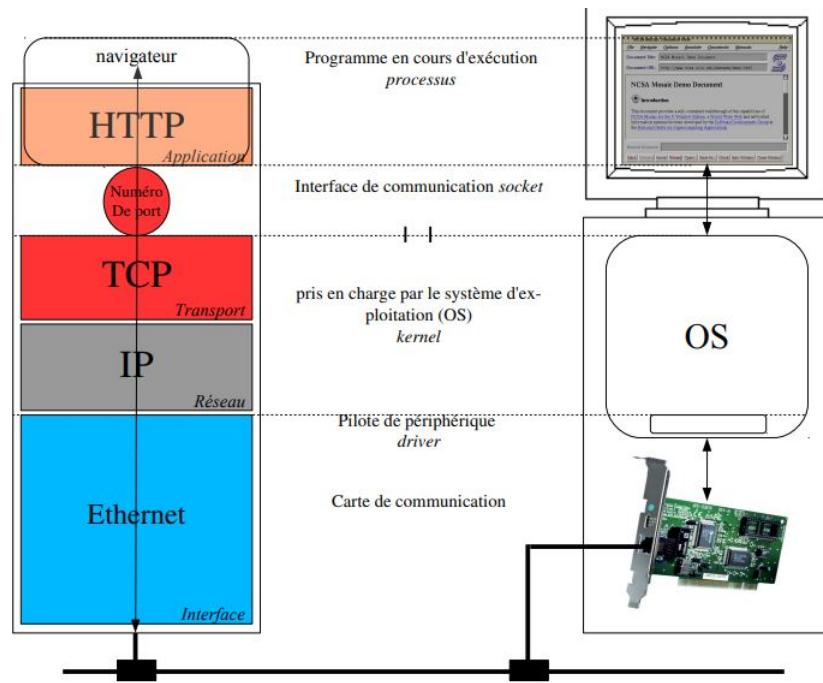
TCP/IP est très répandu, car très robuste !

9.4.1 Vue en couche du modèle TCP/IP

TCP/IP est antérieur à la normalisation OSI et ses couches ne correspondent pas tout à fait :

Applications	couches 7,6 et 5	Protocoles HTTP,FTP,DNS,SMTP,POP	
Transport	couches 4	Protocoles TCP , UDP	
Réseaux	couche 3	Protocoles IP	Routers
Interface	couche 1 et 2	Ethernet, Wifi	Switch, hub

Le modèle TCP/IP



9.4.2 Principe du protocole TCP/IP

Les protocoles TCP et IP permet à tout appareil connecté de dialoguer avec une autre machine connectée.

Le principe du protocole TCP/IP est similaire à celui d'une expédition de colis par la poste. Si je dois envoyer un meuble en kit, il faut commencer par le démonter et d'emballer chaque pièce. Ensuite, il faut porter chaque paquet au bureau de poste le plus proche. Celui-ci va transmettre ces paquets au bureau de poste principal. Les paquets pourront prendre ensuite des chemins différents (suivant la taille, les disponibilités des trains, avions ...) Comme la notice finale est jointe à chaque paquet, on pourra vérifier qu'il ne manque pas de colis (et faire le cas échéant une réclamation), et il sera aisément de monter le meuble.

C'est la même chose sur internet ! Si j'envoie un message, celui-ci est décomposé en petits paquets (qui contiennent des blocs de données). C'est le rôle du protocole TCP. Les différents paquets vont transiter par des routeurs différents jusqu'à la destination finale. Chaque paquet peut avoir une route différente. C'est le rôle du protocole IP de les faire arriver au bon destinataire, et celui-ci pourra connaître l'expéditeur. À l'arrivée, l'ensemble des paquets permettent de reconstituer le message envoyé, grâce au protocole TCP.

Ainsi, et pour simplifier :

- Le protocole TCP se charge de découper le message en petits paquets et au final de reconstituer le message
- le protocole IP ajoute au paquets l'adresse du destinataire et de l'expéditeur. Cela permet au destinataire d'envoyer des accusés de réception pour chaque paquet. L'expéditeur peut éventuellement renvoyer des paquets qui se seraient perdus.

9.4.3 L'adressage

Il existe deux façons d'identifier une machine sur un réseau : l'adressage physique et l'adressage IP.

Remarque

Afin de trouver l'adresse IP locale et l'adresse MAC de votre ordinateur, il suffit de taper ipconfig/all (Windows) dans l'invite de commande, ou bien ifconfig (Linux).

On distinguera donc plusieurs types d'adresse suivant les couches :

Application	Nom, URL, @email
Transport	Numéro de port
Réseau	Adresse réseau
Interface	Adresse physique

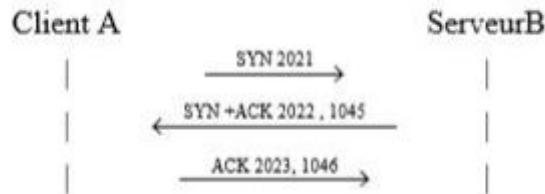
Chaque machine possède une adresse physique unique (MAC). Dans un réseau local, une machine ne peut envoyer une information à une autre que par le biais des adresses physiques. Il faut donc un système qui permet de passer des adresses IP aux adresses MAC : le protocole ARP.

Le protocole RARP permet, quant à lui, de passer des adresses MAC au adresses IP.

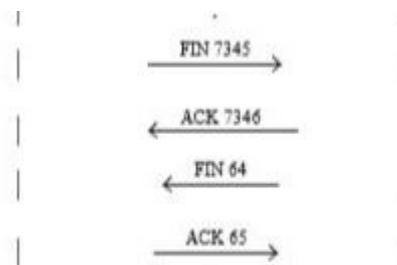
9.4.4 début et fin d'une communication

Comment se déroule la connexion entre deux machines ?

- Début de la connexion : Le client A envoie une demande de synchronisation SYN accompagnée d'un nombre tiré au hasard (ici 2021). Le serveur B répond ACK (acknowledge, soit d'accord) , en augmentant 2021 à 2022 et en rajoutant un nombre tiré au hasard (ici 1045). Le client répond ACK en augmentant les deux nombres.



- La fin de connexion se fait avec des numéros de contrôle également. Ce mécanisme permet au serveur d'envoyer des données qui manquent éventuellement.



9.4.5 Les différentes couches du modèle TCP/IP

Couche Applications

Elles gèrent l'exécution des différentes applications réseaux via une multitude de protocoles (HTTP, HTTPS pour le web, SMTP/POP3/IMAP pour la messagerie électronique, FTP pour le transfert de fichier, DNS pour traduire les noms de domaine en adresse IP...).

Couche transport

Les deux fonctions principales de la couche transport sont :

- Découper des données, de taille variable, en paquets de taille fixe.
- Identifier les programmes destinataire et émetteur de la donnée.

Les protocoles de la couche transport les plus connus sont les protocoles TCP et UDP.

Quand un ordinateur reçoit un paquet, il doit savoir à quel programme est destiné ce paquet (navigateur, web, messagerie, x vidéo). Pour cela, on a défini des ports logiciels : ce sont des numéros, que chaque application va réserver en émettant des données.

Définition

Le port permet au système installé (Linux, Windows) de déterminer à quelle application les données venant du réseau sont destinées.

Pour comprendre ce qu'est un port logiciel (à ne pas confondre avec un port matériel, type USB ou autre), on peut faire une analogie avec le courrier. Quand quelqu'un envoie une lettre, il ne précise pas seulement l'adresse postale, mais aussi la personne à laquelle elle est destinée, au cas où plusieurs personnes vivent à la même adresse.

Ports logiciels courants :

21 : File Transfer Protocol (FTP), 22 : Secure Shell (SSH), 25 : Simple Mail Transfer Protocol (SMTP), 53 : Domain Name System (DNS) service, 80 : Hypertext Transfer Protocol (HTTP), 110 : Post Office Protocol (POP3), 143 : Internet Message Access Protocol (IMAP), 443 : HTTP Secure (HTTPS)

Le paquet créé par la couche transport est appelé TCP a pour format simplifié :

Port source	Port dest	N SYN	N ACK	Long	Données
-------------	-----------	-------	-------	------	---------

Définition

Une socket est un point de communication par lequel un processus peut émettre et recevoir des informations. Ce point de communication devra être relié à une adresse IP et un numéro de port.

Couche réseau

Un des protocoles de la couche réseau est le protocole IP.

La première étape est de déterminer si le destinataire et la source sont sur le même réseau. Dans le cas où la source et le destinataire se trouvent sur des réseaux différents il faut passer par un routeur, et le paquet IP peut ainsi traverser de nombreux routeurs avant d'atteindre sa destination. Chaque routeur (et hôte) possède une table de routage et en fonction de l'adresse destination, il choisit l'entrée qui correspond le mieux et émet de nouveau le paquet sur l'interface en sortie correspondante.

Définition

| Un routeur est un ordinateur dont la seule fonction est d'acheminer des informations sur le réseau.

Définition

| Un hôte est un ordinateur qui n'est pas un routeur

Les segments sont encapsulés dans des paquets IP qui contiennent en plus les adresses logiques de la source et du destinataire, la longueur du segment, sa durée de vie. On teste si la source et le destinataire sont sur le même réseau, sinon la source au niveau de cette couche devient un routeur.

Ainsi, et pour simplifier :

- le protocole IP ajoute au paquets l'adresse du destinataire et de l'expéditeur. Cela permet au destinataire d'envoyer des accusés de réception pour chaque paquet. L'expéditeur peut éventuellement renvoyer des paquets qui se seraient perdus.

Les paquets envoyés par le protocole IP sont appelés datagrammesIP et ont pour format simplifié :

En-tête	Adresse IP source	Adresse IP dest	Segment TCP
---------	-------------------	-----------------	-------------

Les données sont donc à l'intérieur du paquet TCP qui est lui même dans le paquet IP.

Couche interface

La gestion des réseaux locaux se fait sur la couche liaison, qui prend en charge la mise en forme des données, l'identification des ordinateurs et la détection, et correction des erreurs de transmission (réseau Ethernet ou Wifi par exemple).

Le protocole Ethernet envoie des trames sur le réseau, dans ce format simplifié :

Adresse MAC DST	Adresse MAC SRC	DatagrammeIP	Code de correction
-----------------	-----------------	--------------	--------------------

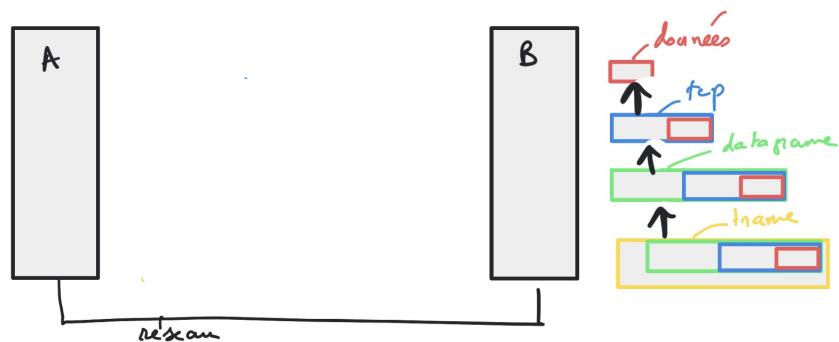
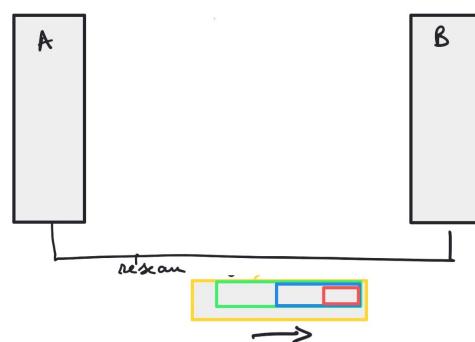
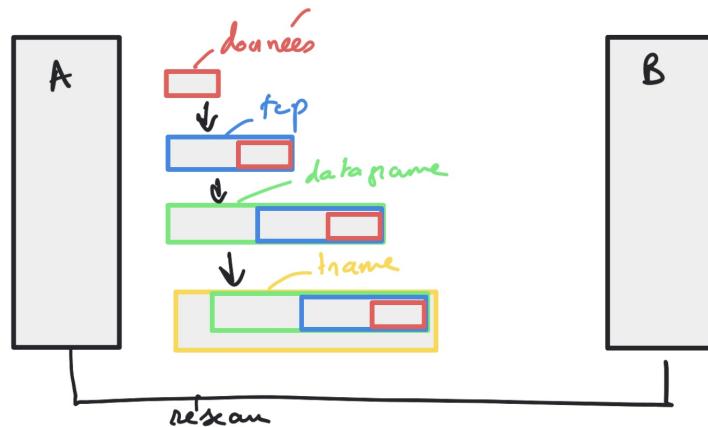
On constate que le paquet IP est dans la trame Ethernet.

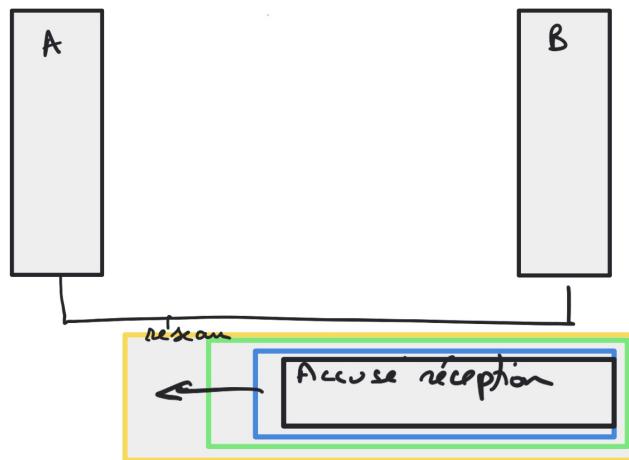
Finalement, les données sont dans le paquet TCP, qui est dans le paquet IP , qui est lui même dans la trame Ethernet : on parle d' encapsulation.

9.4.6 Accusé de réception

Le protocole TCP permet de s'assurer qu'un paquet est bien arrivé à destination. En effet quand l'ordinateur B reçoit un paquet de données en provenance de l'ordinateur A, l'ordinateur B envoie un accusé de réception à l'ordinateur A (un peu dans le genre "OK, j'ai bien reçu le paquet"). Si l'ordinateur A ne reçoit pas cet accusé de réception en provenance de B, après un temps prédéfini, l'ordinateur A renverra le paquet de données vers l'ordinateur B.

Nous pouvons donc résumer le processus d'envoi d'un paquet de données comme suit :

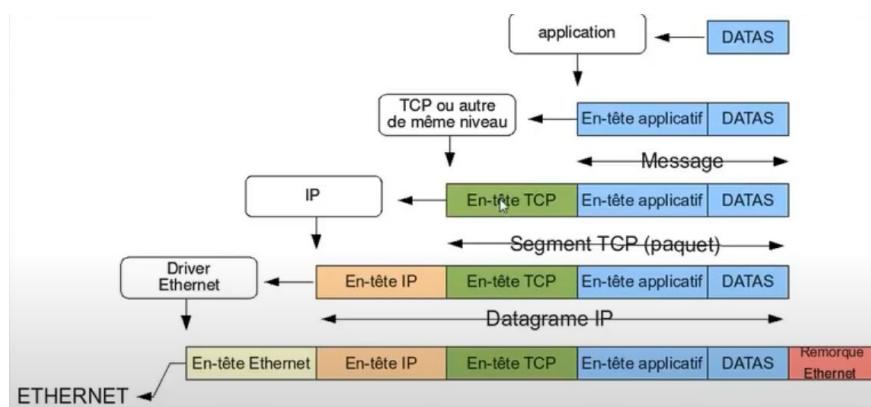




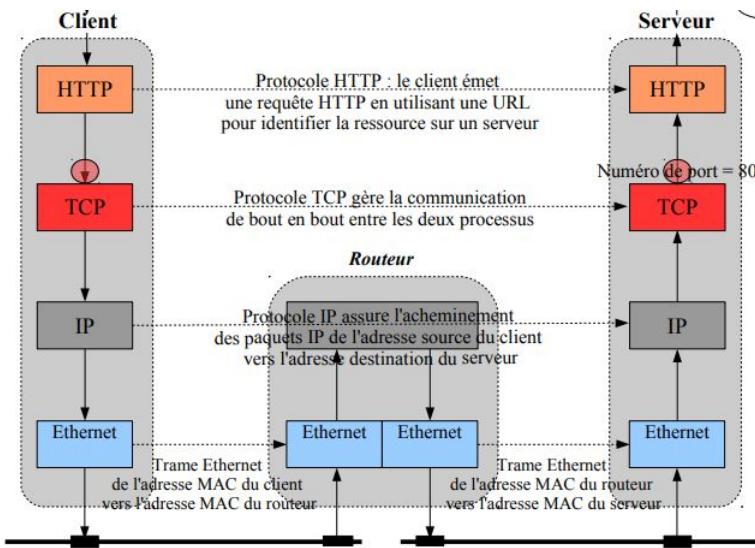
9.4.7 Encapsulation

L'encapsulation consiste à envelopper les données à chaque couche du modèle TCP/IP.

L'encapsulation



Protocoles à l'œuvre lors du routage



9.4.8 Observer le réseau avec Wireshark

Wireshark est un logiciel gratuit qui permet d'observer tout ce qui circule sur les réseaux. Il permet également d'observer le contenu de chaque couche, le nom des protocoles, les en-têtes, les flags...

Exercice 9.159

Voici un extrait d'une capture avec le logiciel Wireshark :

Wireshark · Paquet 81 · Ethernet

```
> Frame 81: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface \Device\NPF_{25404A0
> Ethernet II, Src: Giga-Byt_6b:85:9a (1c:1b:0d:6b:85:9a), Dst: Iskratel_b5:7e:ab (64:6e:ea:b5:7e:ab)
> Internet Protocol Version 4, Src: 192.168.0.26, Dst: 172.217.20.133
> Transmission Control Protocol, Src Port: 51122, Dst Port: 443, Seq: 2804, Ack: 1184, Len: 0
```

0000	64 6e ea b5 7e ab 1c 1b	0d 6b 85 9a 08 00 45 00	dn ~ k E
0010	00 28 df 2f 40 00 80 06	99 7f c0 a8 00 1a ac d9	(/@
0020	14 85 c7 b2 01 bb 8e 74	d4 f1 4f 66 c3 d7 50 10 t .of ..P
0030	0f fb dd a6 00 00		

1. Analyse d'une trame

- Quel est le format utilisé : IPv4 ou IPv6 ?
- Quelle est l'adresse source ? Est-ce une adresse publique ou privée ? S'il s'agit d'une adresse publique, sur quel site pointe-t-elle ?
- Quelle est l'adresse destinataire ? Est-ce une adresse publique ou privée ? S'il s'agit d'une adresse publique, sur quel site pointe-t-elle ?
- Quel est le protocole utilisé par la couche Application ?
- Comment expliquer cette numérotation ?

0000
0010
0020
0030

2. Somme de contrôle :

On désire maintenant considérer le paquet Ip de la couche 3 (datagramme) et faire une vérification sommaire de l'intégrité du paquet avec la "somme de contrôle".

Voici en surbrillance le paquet IPv4 au format hexadécimal.

```
> Frame 83: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface
> Ethernet II, Src: Iskratel_b5:7e:ab (64:6e:ea:b5:7e:ab), Dst: Giga-Byt_6b:8
> Internet Protocol Version 4, Src: 172.217.20.133, Dst: 192.168.0.26
> Transmission Control Protocol, Src Port: 443, Dst Port: 51122, Seq: 1184, A
<

0000  1c 1b 0d 6b 85 9a 64 6e ea b5 7e ab 08 00 45 00 . . . k . dn . . ~ . . E .
0010  00 28 7a 64 00 00 79 06 45 4b ac d9 14 85 c0 a8 . . . (zd . . y . EK . . .
0020  00 1a 01 bb c7 b2 4f 66 c3 d7 8e 74 d5 18 50 10 . . . . . Of . . . t . . P .
0030  08 c0 e4 ba 00 00 d6 9e 00 00 00 00 . . . . . . . . . . . . . . . . . . . . . .
```

Il y a donc 20 octets.

- Regrouper ces paquets pour faire 10 paquet de 2 octets.
 - Effectuer en hexadécimal la somme des 5 premiers groupes et des 4 derniers (on exclut donc le 6ème groupe).
 - Comme le nombre dépasse 2 octets, on additionne le caractère du poids fort (le premier chiffre) avec le nombre formé des 4 plus faibles. Qu'obtient-on ?
 - Écrire ce nombre en binaire.
 - Écrire son complément à 2 et le convertir en hexadécimal.
 - La somme de contrôle est correct si elle est égale au 6ème paquet. Est-elle correcte ?
3. Créer une fonction python qui :
- a pour paramètre une chaîne de caractère de 40 caractères, correspondant à la trame IP.
 - renvoie True si la somme de contrôle est correcte, False sinon.

Exercice 9.160

Sur un ordinateur connecté à un sous-réseau du lycée, le résultat de la commande ipconfig fournit les renseignements ci-dessous :

Adresse IP . . . : 192.168.22.45

Masque : 255.255.248.0

Adresse MAC. . . : 00 :24 :1D :B3 :D3 :3C

Passerelle . . . : 192.168.16.254.

- Quel protocole permet de traduire l'adresse MAC d'une interface réseau en adresse IP ?
- Expliquer pourquoi la notation CIDR de l'adresse IP de cet ordinateur s'écrit 192.168.22.45/21.
- Déterminer l'adresse du sous-réseau auquel est connecté cet ordinateur.
- Dans la plage des adresses possibles dans ce sous-réseau, l'adresse de diffusion est la dernière. Déterminer l'adresse de diffusion de ce sous-réseau.
- Combien d'appareils au plus peuvent être simultanément connectés à ce sous-réseau ?
- À son domicile, un élève teste la connexion à cet ordinateur à l'aide de la commande ping 192.168.22.45 : pourquoi n'obtient-il pas de réponse ?