TP Réseau 1

I Logiciel de simulation Filius

Télécharger le logiciel de simulation réseau : Filius
Lien de téléchargement : https://www.lernsoftware-filius.de/Herunterladen (Le site web et l'installateur sont en allemand, mais le logiciel est traduit en français).

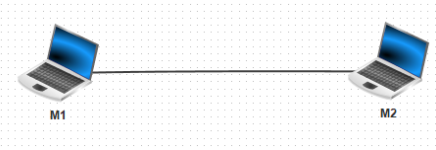
- Documentation complète (en anglais) : http://www.lernsoftware-filius.de/downloads/Introduction_Filius.pdf

- Attention : Choisir la langue lors de la première ouverture du logiciel.

Regardez cette vidéo qui vous montre quelques fonctionnalités de Filius :
https://www.youtube.com/watch?v=nzuRSOwdF5I

Il est possible de faire communiquer deux ordinateurs en les reliant par un simple câble. On dit alors que ces deux ordinateurs sont en réseau.

Créons ce réseau basique avec Filius. Ouvrez Filius et construisez ce réseau :



Changez les noms et attribuez-leur les adresses ip :

192.168.0.1 et 192.168.0.2

(clic droit et configurer)

Vérifions que ces 2 ordinateurs peuvent communiquer. Passez en mode communication (triangle vert) . Clic sur M1. Clic sur " Installez des logiciels " et installez la ligne de commande. Tapez (après root />) ping 192.168.0.2. Les paquets sont bien transmis et reçus : la communication est établie.

II Matériel et réseau

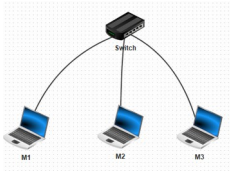

prises RJ 45

Dans la plupart des cas, le câble reliant les 2 ordinateurs est un câble **Ethernet**. Ce type de câble possède à ses 2 extrémités **des prises RJ45**.
Un ordinateur relié à un réseau doit posséder **une carte réseau**, on identifie cette carte réseau de type Ethernet grâce à la prise RJ45 femelle située souvent à l'arrière de l'ordinateur. 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


switch

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.

Etendre le réseau précédent de cette façon :



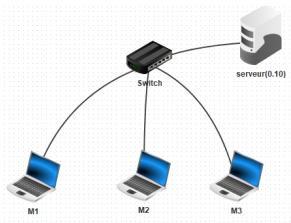
Attribuez l'adresse IP : 192.168.0.3 à M3

Vérifiez que M1 peut communiquer avec M3 en lançant un ping.

Une adresse **IP** (Internet Protcol) permet d'identifier une machine (plus précisément une interface de cette machine) dans un réseau d'ordinateurs. L'adresse IP permet **le routage** des paquets de données d'un ordinateur à l'autre. Les nombres que contient l'adresse IP permettent en effet aux paquets de trouver leur chemin de nœud en nœud depuis l'adresse IP source jusqu'à l'adresse IP destinataire.

III Rôle d'un serveur dans un réseau

1) Ajout dans un réseau



Ajouter un serveur. On choisira un PC. On le renomme en serveur(0.10).

On rappelle qu'un serveur est un ordinateur qui répond à des requêtes émises par des clients (ici M1,M2,M3).

Passez en mode communication. Cliquez sur serveur. Installez " serveur générique ". Ouvrez " serveur générique " et démarrez-le. Ainsi, notre serveur est prêt à fonctionner (seulement pour recevoir et réemettre des messages pour l'instant).

2) Envoi d'un message du client au serveur

Envoyons un message depuis M1 vers notre serveur. Pour cela, installez dans M1 un client générique.
Ouvrez l'application " Client générique ". Entrez l'adresse IP du serveur 192.168.0.10. Connectez-vous au serveur et écrire un message. Envoyez votre message (" Coucou! ") et regardez dans la fenêtre du serveur les infos indiquées : le serveur a bien reçu le message et l'a renvoyé.
Cliquez droit sur le serveur et faire : afficher les échanges de données (192.168.0.10).

3) Analyse des messages

a) adresse MAC et protocole ARP

Lorsque le client (M1) s'est **connecté** au serveur, voici les paquets qui ont été envoyés :

serveur(0.10) - 192.168.0.10

No.	Date	Source	Destination	Protocole	Couche	Commentaire
1	14:42:14.392	192.168.0.1	192.168.0.10	ARP	*Internet*	Recherche de l'adresse MAC associée à 192.168.0.10, 192.168.0.1: 01:AA:DC:0C:0C:04
2	14:42:14.394	192.168.0.10	192.168.0.1	ARP	*Internet*	192.168.0.10: 87:AF:FB:88:B2:5D
3	14:42:14.600	192.168.0.1:55937	192.168.0.10:55555	TCP	*Transport*	SYN, SEQ: 2148916175
4	14:42:14.603	192.168.0.10:55555	192.168.0.1:55937	TCP	*Transport*	SYN, ACK:2148916176, SEQ: 1361560741
5	14:42:14.809	192.168.0.1:55937	192.168.0.10:55555	TCP	*Transport*	ACK: 1361560742

La ligne 1 correspond à l'envoi d'un paquet du client vers le serveur. Dans ce message, le client donne son adresse mac (01:AA:DC:0C:0C:04) et demande celle du serveur. Le paquet a été envoyé selon le protocole **ARP** (Address Resolution Protocol) au niveau de la couche Internet du modèle **OSI** (Open Systems Interconnexion: voir explications ensuite). Cela veut dire que le serveur, pour répondre à cette requête, va écrire dans sa mémoire au sein de sa table ARP la correspondance entre l'adresse IP de M1 et son adresse mac. Puis, il va renvoyer selon le même protocole, son adresse mac (87:AF:FB:88:B2:5D). Dans la ligne 2, le client reçoit l'adresse mac du serveur et la copie dans sa table ARP.

Une table ARP est la liste enregistrée dans la mémoire d'un ordinateur relié à un réseau des correspondances entre les adresses IP et les adresses MAC des ordinateurs avec lesquelles il a échangé des informations.

Vous pouvez facilement vérifier que le client M1 a mis à jour sa table ARP en tapant: arp en ligne de commande. Vous devriez obtenir ceci (vous devez toujours avoir votre client connecté):

```
root /> arp
|    Adresse IP     |     Adresse MAC     |
-------------------------------------------
| 255.255.255.255  | FF:FF:FF:FF:FF:FF |
| 192.168.0.10      | 87:AF:FB:88:B2:5D |
-------------------------------------------
```

Les adresses mac sont gravées sur les cartes réseaux. Elles permettent d'identifier un ordinateur dans un réseau local. Elles ne permettent pas le routage des paquets d'informations à travers plusieurs réseaux.

b) Mise en place d'une connexion client-serveur dans le protocole TCP (voir explications à la fin du TP)

Les lignes 3,4 et 5 des échanges d'informations correspondent à une mise en place d'une connexion par un système d'accusés de réceptions du protocole TCP. C'est en processus en trois temps (three way handshake). Cela permet d'être cerrtain que la connexion fonctionne bien dans les 2 sens :

- ligne 3 : le client envoie au serveur un paquet (SYN : synchronisation) avec une séquence denombre choisi aléatoirement ici : 2148916175
- ligne 4 : le serveur envoie une reconnaissance (ACKnowlegment) en renvoyant le nombre incrémenté de 1 : 2148916176 avec un autre nombre choisi aléatoirement (SEQ) : 1361560741
- ligne 5 : le client renvoie au serveur son nombre signe de reconnaissance (ACK)

c) couche application du modèle OSI

Lorsque le client a envoyé le message " Coucou ! " au serveur, les informations suivantes ont transité sur le réseau :

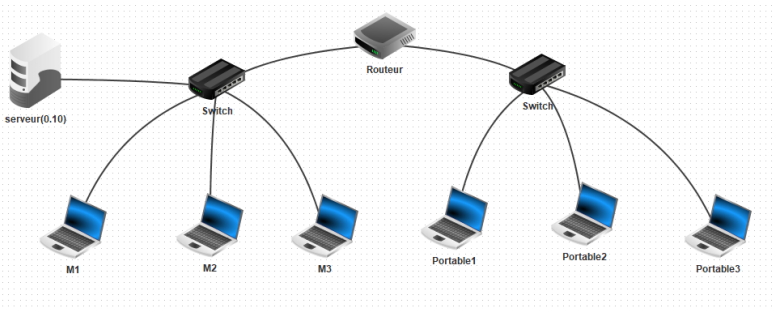
11	15:57:23.885	192.168.0.1:17764	192.168.0.10:55555		*Application*	coucou !
12	15:57:23.891	192.168.0.10:55555	192.168.0.1:17764	TCP	*Transport*	ACK: 2012228262
13	15:57:23.945	192.168.0.10:55555	192.168.0.1:17764		*Application*	coucou !
14	15:57:24.171	192.168.0.1:17764	192.168.0.10:55555	TCP	*Transport*	ACK: 613825519

Remarquons simplement que la transmission du message s'effectue au niveau de la **couche Application** dans le modèle **OSI** puisqu'elle est un échange entre 2 applications (client et serveur). A noter aussi que le serveur renvoie un " acknowledgment " (un accusé de réception) par TCP et le message. Le client renvoie une deuxième fois le message. Voilà pourquoi le protocole TCP est plus lent que l'UDP.

IV Ajout d'un second réseau

Mode conception :
Ajoutons un second réseau local avec 3 nouveaux ordinateurs comme ci-dessous. Nommons-les avec des IP allant de 192.168.1.1 à 192.168.1.3

Mettez 2 interfaces au routeur. Remarquez que ce deuxième sous-réseau à une adresse IP différente au niveau du troisième nombre : 1 à la place de 0 pour le premier sous-réseau. C'est ainsi qu'on différencie les sous réseaux entre eux.



V Rôle d'un routeur

Mode conception :
Connectons les 2 réseaux à l'aide d'un routeur dont les cartes d'interface seront configurées avec les switchs avec les adresses IP 192.168.0.20 et 192.168.1.20

Mode simulation :
Testons la connexion entre les postes 192.168.0.1 et 192.168.1.1 avec la commande PING. On obtient ceci :

C'est normal puisque les paquets doivent passer par le routeur : il faut dire que le routeur est la passerelle entre les deux sous-réseaux.

```
root /> ping 192.168.1.1
Destination inaccessible
```

Rectifions cette erreur :
- pour chacun des 3 ordinateurs de gauche (serveur compris), indiquez 192.168.0.20 pour la passerelle.
- pour chacun des 3 ordinateurs de droite, indiquez 192.168.1.20

Testons à nouveau la connexion entre les postes 192.168.0.1 et 192.168.1.1 avec la commande PING. Ça fonctionne :

```
root /> ping 192.168.1.1
PING 192.168.1.1 (192.168.1.1)
From 192.168.1.1 (192.168.1.1): icmp_seq=1 ttl=63 time=4822ms
From 192.168.1.1 (192.168.1.1): icmp_seq=2 ttl=63 time=2408ms
From 192.168.1.1 (192.168.1.1): icmp_seq=3 ttl=63 time=2408ms
From 192.168.1.1 (192.168.1.1): icmp_seq=4 ttl=63 time=2407ms
--- 192.168.1.1 Statistiques des paquets ---
4 paquets transmis, 4 paquets reçus, 0% paquets perdus
```

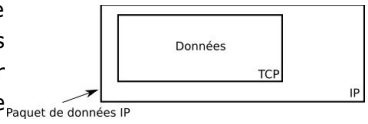
VI Le protocole TCP/IP

En 1974 Vint Cerf et Bob Khan vont mettre au point le protocole TCP qui sera très rapidement couplé au protocole IP pour donner TCP/IP. TCP/IP, grâce à sa simplicité, va très rapidement s'imposer comme un standard : les différents réseaux (ARPAnet et les autres) vont adopter TCP/IP. Cette adoption va permettre d'interconnecter tous ces réseaux (2 machines appartenant à 2 réseaux différents vont pouvoir communiquer grâce à cette interconnexion). Internet était né (le terme Internet vient de "internetting" qui signifie "Connexion entre plusieurs réseaux"). TCP/IP est donc au coeur d'Internet, voilà pourquoi aujourd'hui, la plupart des machines utilisent TCP/IP.

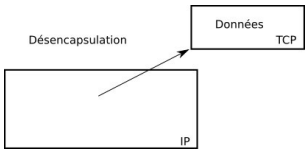
Après ce petit rappel historique, essayons de comprendre le principe de base des protocoles TCP (Transmission Control Protocol) et IP (Internet Protocol)

Quand un ordinateur A "désire" envoyer des données à un ordinateur B, l'ordinateur A "utilise" le protocole TCP pour mettre en forme les données à envoyer.

Ensuite le protocole IP prend le relai et utilise les données mises en forme par le protocole TCP afin de créer des paquets des données. Après quelques autres opérations qui ne seront pas évoquées ici, les paquets de données pourront commencer leur voyage sur le réseau jusqu'à l'ordinateur B. Il est important de bien comprendre que le protocole IP "encapsule" les données issues du protocole TCP afin de constituer des paquets de données.

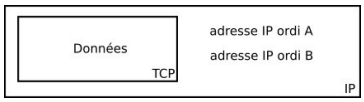


Une fois arrivées à destination (ordinateur B), les données sont "désencapsulées" : on récupère les données TCP contenues dans les paquets afin de pouvoir les utiliser.



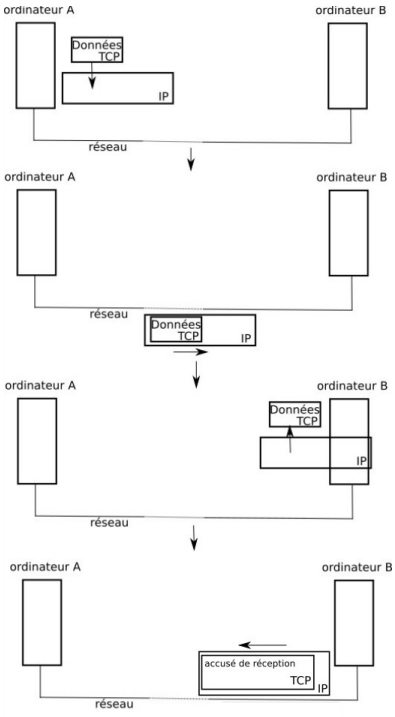
Le **protocole IP** s'occupe uniquement de faire arriver à destination les paquets en utilisant l'adresse IP de l'ordinateur de destination. Les adresses IP de l'ordinateur de départ (ordinateur A) et de l'ordinateur destination (ordinateur B) sont ajoutées aux paquets de données.

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" : un ACK). 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 (ci-dessous à gauche) :



À noter qu'il existe aussi **le protocole UDP** qui ressemble beaucoup au protocole TCP. La grande différence entre UDP et TCP est que le protocole UDP ne gère pas les accusés de réception. Les échanges de données avec UDP sont donc moins fiables qu'avec TCP (un paquet "perdu" est définitivement "perdu" et ne sera pas renvoyé) mais beaucoup plus rapides (puisqu' il n'y a pas d'accusé de réception à transmettre). UDP est donc très souvent utilisé pour les échanges de données qui doivent être rapides, mais où la perte d'un paquet de données de temps en temps n'est pas un gros problème (par exemple le streaming vidéo).

Il est très important de bien comprendre que **TCP/IP repose sur la notion de paquets de données**. Si par exemple on désire envoyer un fichier (son, photo, vidéo ou texte, peu importe, dans tous les cas on envoie une succession de bits) en utilisant TCP/IP, les données qui constituent ce fichier ne seront pas envoyées d'un seul tenant, ces données vont être "découpées" en plusieurs morceaux et chaque morceau sera envoyé dans un paquet différent. Une fois tous les paquets arrivés à destination, le fichier d'origine pourra être reconstitué. Pour aller d'un ordinateur A à un ordinateur B, les différents paquets contenant les données qui constituent notre fichier, ne passeront pas forcement par la même route (cette notion de route sera abordée plus tard), ils pourront emprunter des chemins très différents : en exagérant à peine, pour faire le trajet Paris-Los Angeles, certains paquets pourront passer par l'atlantique alors que d'autres passeront par le pacifique. Si un des paquets

n'arrive pas à destination, le fichier ne pourra pas être reconstitué, le paquet "perdu" devra être renvoyé par l'émetteur (voir le système d'accusé de réception décrit ci-dessus)

Le protocole TCP fait partie de la couche Transport .

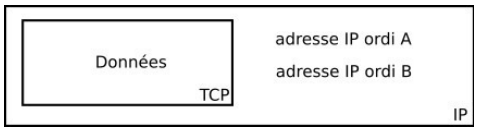
VII Protocoles réseaux

1) Encapsulation

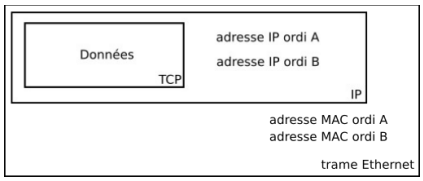
Nous avons eu l'occasion de voir avec les protocoles TCP et IP le processus d'encapsulation des données : "IP encapsule TCP". Les paquets IP ne peuvent pas transiter sur un réseau tel quel, ils vont eux aussi être encapsulés avant de pouvoir "voyager" sur le réseau. L'encapsulation des paquets IP produit ce que l'on appelle une trame. Il n'est pas question d'étudier en détail ce qu'est une trame, vous devez juste savoir qu'il existe de nombreux types de trames : ATM, token ring, PPP, Ethernet, Wifi... Nous allons uniquement évoquer les 2 dernières : la trame Ethernet et la trame Wifi.

Si vous utilisez un réseau filaire avec des câbles Ethernet (avec des prises RJ45), la trame sera de type Ethernet (ce qui est le cas pour le réseau du lycée). Si vous utilisez un réseau sans fil Wifi, la trame sera de type Wifi. En faite, la trame Wifi ressemble beaucoup à la trame Ethernet, on peut même dire que la trame Wifi est la variante sans-fil de la trame Ethernet, afin de simplifier les choses, dans la suite, nous évoquerons uniquement la trame Ethernet en ayant à l'esprit que ce qui est dit sur la trame Ethernet et aussi valable pour la trame Wifi.

Nous avons vu que le paquet IP contient les adresses IP de l'émetteur et du récepteur :



Le paquet IP étant encapsulé par la trame Ethernet, les adresses IP ne sont plus directement disponibles (il faut désencapsuler le paquet IP pour pouvoir lire ces adresses IP), nous allons donc trouver un autre type d'adresse qui permet d'identifier l'émetteur et le récepteur : l'adresse MAC (Media Access Control) aussi appelée adresse physique.



2) Adresse MAC

Une adresse MAC est codée sur 6 octets. on écrit traditionnellement les adresses MAC en hexadécimal, chaque octet étant séparés par 2 points (exemple d'adresse MAC : 00:E0:4C:68:02:11)

L'adresse MAC est liée au matériel, chaque carte réseau (Ethernet ou Wifi) possède sa propre adresse MAC, il n'existe pas dans le monde, 2 cartes réseau (Ethernet ou Wifi) qui possèdent la même adresse MAC. Les 3 premiers octets d'une adresse MAC ("00:E0:4C" dans l'exemple ci-dessus) désignent le constructeur du matériel, par exemple, "00:E0:4C" désigne le constructeur "realtek semiconductor corp".

Au moment de l'encapsulation d'un paquet IP, l'ordinateur "émetteur" va utiliser un protocole nommé ARP (Address Resolution Protocol) qui va permettre de déterminer l'adresse MAC de l'ordinateur "destination", en effectuant une requête "broadcast" (requête destinée à tous les ordinateurs du réseau) du type : "j'aimerais connaitre l'adresse MAC de l'ordinateur ayant pour IP XXX.XXX.XXX.XXX". Une fois qu'il a obtenu une réponse à cette requête ARP, l'ordinateur "émetteur" encapsule le paquet IP dans une trame Ethernet et envoie cette trame sur le réseau.

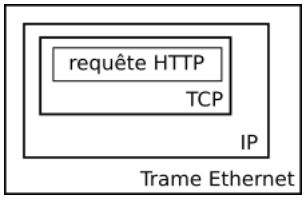
3) Couche application

Nous avons vu que le protocole TCP permet de mettre en forme les données à envoyer :

Quelle est la nature de ces données mises en forme par TCP ?

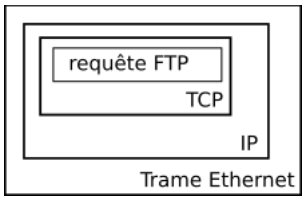
En fait, TCP effectue lui aussi une encapsulation, les données encapsulées par TCP peuvent être de plusieurs natures :

Nous avons étudié le protocole HTTP. Les requêtes et les réponses HTTP sont encapsulés par TCP, au bout du compte et en résumé, nous avons donc :



TCP encapsule aussi d'autres types de requêtes (et réponses), par exemple FTP (File Transfer Protocol) qui permet d'envoyer sur un réseau des fichiers (texte, son, image...), SMTP (Simple Mail Transfer Protocol) qui permet d'envoyer des emails, DNS (Domain Name Server) qui permet d'avoir la correspondance entre une adresse IP et une URL (prochain TP),...

Il est donc aussi possible d'avoir :



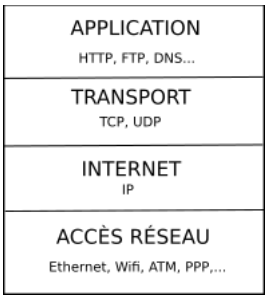
On dit que tous ces protocoles (HTTP, FTP, SMTP, DNS,...) appartiennent à la couche "Application" du modèle TCP/IP.

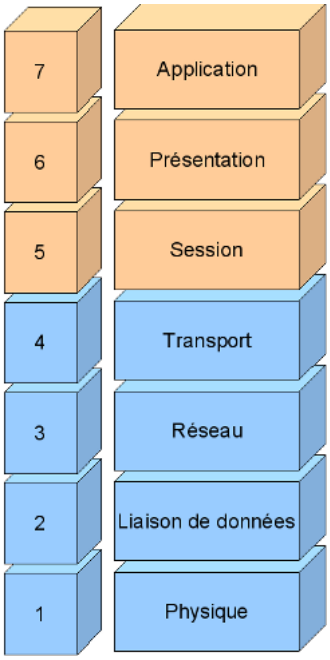
4) Le modèle des couches TCP/IP

En effet, à chaque phase d'encapsulation on associe ce que l'on appelle une couche :

- comme nous l'avons vu les protocoles HTTP, FTP, SMTP, DNS,... sont associés à la couche "Application"
- les protocoles TCP et UDP sont associés à la couche "Transport"
- le protocole IP est associé à la couche "Internet"
- les trames Ethernet (ou Wifi) sont associées à la couche "Accès réseau"

On présente souvent ces différentes couchent sur ce type de schéma :





La couche du "dessous" encapsule la couche située "au dessus". On nomme ce système de couche "modèle de couches TCP/IP" (car ce modèle repose principalement sur TCP et IP)

5) Le modèle des couches OSI

Il existe un autre modèle de couche, le modèle OSI (Open Systems Interconnection), ce système est antérieur au modèle TCP/IP puisqu'il date des années 1970. Ce modèle est principalement théorique et à permis de poser les bases des communications réseau. Ce modèle est composé de 7 couches (alors que le modèle TCP/IP est composé de 4 couches).

Il existe des liens entre le modèle OSI et le modèle TCP/IP (par exemple on retrouve le protocole IP dans la couche 3 du modèle OSI et TCP dans la couche 4), mais parfois comparer les 2 modèles peut être délicat.

Ce modèle est donné ici à titre d'information (pour le cas où vous le rencontriez pendant vos recherches sur Internet), mais le principal est de retenir ce qui a été vu sur le modèle TCP/IP.