Architectures Matérielles et Systèmes d'Exploitation

Protocoles de communication

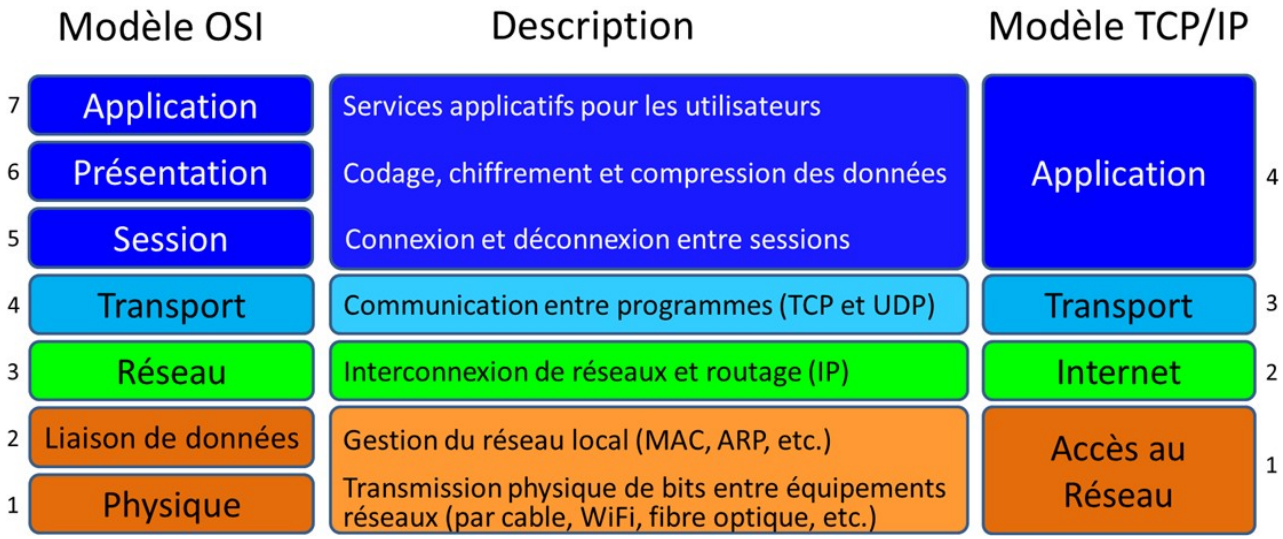
Les modèle OSI et TCP/IP

Pour que des machines communiquent sur un réseau, c'est-à-dire se transmettent des informations, il ne suffit pas qu'elles parlent un langage commun, il faut aussi qu'elles partagent des règles d'émission et de réception des données contenant ces informations. C'est le rôle d'un protocole.

L'ensemble des **règles** qui permettent à deux machines de communiquer sur un réseau s'appelle un **protocole**.

Afin de s'y retrouver au milieu des protocoles réseaux, les modèles OSI et TCP/IP permettent de les regrouper selon leurs niveaux d'abstraction, dans ce qu'on appelle des « **couches réseaux** »[1] :

- Le **modèle OSI** (de l'anglais *Open Systems Interconnection*) comporte sept couches.
- Le **modèle TCP/IP**, plus simple, comporte quatre couches : Accès au Réseau, Internet, Transport et Application.



La suite de l'activité parcourt les différentes couches réseaux en utilisant le simulateur de réseau Filius [2].

Le logiciel dispose de deux modes ; on passe d'un mode à l'autre en cliquant sur l'icône correspondante :

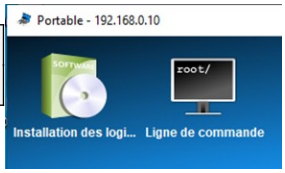
— le mode conception, activé par l'icône « marteau » : .
— le mode simulation, activé par l'icône « flèche verte » : .

Couche « Accès Réseau »

1. En mode conception , ajoutez un premier ordinateur portable.
2. Passez en mode simulation . Un double-clic sur l'ordinateur permet d'ouvrir l'installateur de logiciels. Installez la Ligne de commande en la faisant passer à gauche avec la flèche (ou avec un double clic).

3. Ouvrez la ligne de commande (double-clic) et inspectez la liste des commandes disponibles. Quelle commande permet d'afficher les paramètres du réseau ?

ipconfig



1 Visionner Comprendre les modèles OSI et TCP/IP : https://www.youtube.com/watch?v=26jazyc7VNk
2 https://www.lernsoftware-filius.de/

4. Saisissez cette commande dans l'interpréteur (on reconnaît un shell de type linux). Quels paramètres réseau nous apporte-t-elle sur cet ordinateur ?

L'adresse IP, le Masque, l'adresse MAC, la Passerelle et le Serveur DNS

La commande **`ipconfig`** permet d'afficher la **configuration réseau** d'une machine.

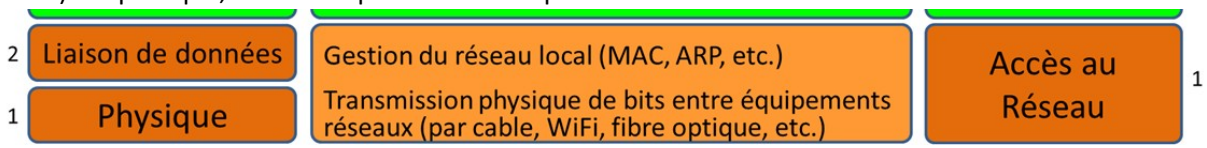
5. Combien cet ordinateur a-t-il d'adresses ?

2 adresses : une adresse MAC et une adresse IP

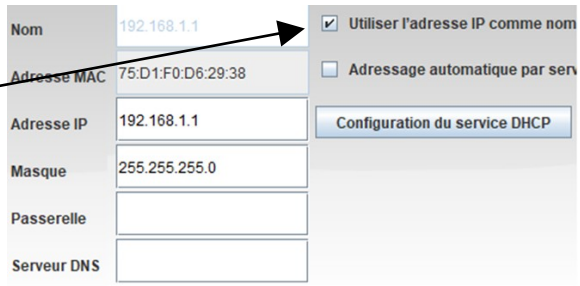
L'**adresse MAC** (*Media Access Control*), parfois nommée **adresse physique**, est un identifiant stocké dans une carte ou interface réseau donné par le constructeur. Elle est **unique au monde**.

L'adresse MAC est constituée de 6 octets écrits sous forme hexadécimale, chacun séparé par " : ", par exemple ici 1B:22:62:47:33:D7. Si une machine possède deux cartes d'interface réseau, par exemple une carte Ethernet et une carte Wi-Fi, alors elle aura deux adresses MAC.

L'adresse MAC est utilisée au niveau de la couche « Accès au Réseau » du modèle TCP/IP (ou des des couches 1 et 2 du modèle OSI). En pratique, on la manipule rarement quand on s'intéresse à un réseau.



6. Retournez en mode conception un double-clic sur l'ordinateur permet d'accéder à sa configuration. Configurez l'ordinateur avec les informations suivantes:
   - Nom : choisissez « Utiliser l'adresse IP comme nom »
   - Adresse MAC : on ne peut pas la modifier
   - Adresse IP : Changez en 192.168.1.1
   - Masque : 255.255.255.0
   - Passerelle et Serveur DNS : laissez vide pour l'instant

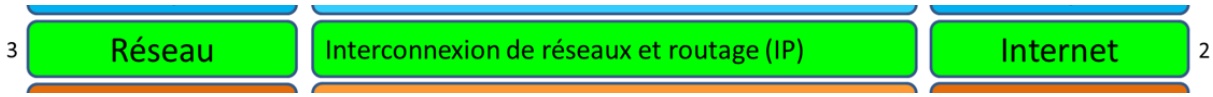


Couche « Internet »

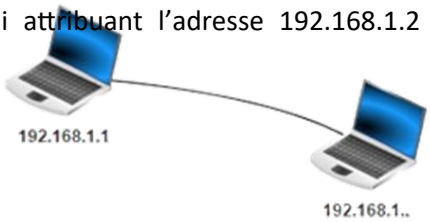
L'adresse **IP** (***Internet Protocol***) identifie une machine à l'intérieur d'un réseau. Elle est **unique sur ce réseau**.
Il existe deux types d'adresse IP : **IPv4** sur 32 bits et **IPv6** sur 128 bits.

Version	**IPv4**	**IPv6**
Description	C'est la version encore la plus utilisée mais avec seulement **32 bits** utilisés, le nombre d'adresses disponibles ($2^{32} \approx 4$ milliards) n'est plus suffisant par rapport aux besoins du monde actuel.	Elle remplace progressivement la version IPv4. Avec **128 bits**, le nombre d'adresses disponibles est de 2^{128}, c'est-à-dire 3,4 x 10^{38} adresses, ce protocole semble donc inépuisable.
Adresse IP	**4 nombres décimaux compris entre 0 et 255 (8 bits) séparés par des point**s	**8 nombres hexadécimaux, compris entre 0000 et ffff (16 bits), écrits avec des chiffres et les lettres de « a » à « f », séparés par des deux-points (:)**.
Exemple	172.16.254.1	2001:0db8:3c4d:0015:0000:0000:1a2f:1a2b

L'adresse IP est utilisée au niveau de la couche 3 du modèle OSI ou couche Internet du modèle TCP/IP.



7. Ajoutez un second ordinateur portable et changez son adresse IP en lui attribuant l'adresse 192.168.1.2 et sélectionnez « Utiliser l'adresse IP comme nom ».



Notez son adresse MAC

95:8C:C3:AA:2E:33

8. Reliez les deux ordinateurs par un câble Ethernet  puis observez qu'un câble posé peut ensuite être supprimé : clic-droit puis « supprimer ».
9. Passez en mode simulation  et ouvrez la ligne de commande du premier ordinateur 192.168.1.1.

Saisissez la commande **`arp`** dans la console. Quelles informations nous apporte cette commande?

```
|     Adresse IP     |    Adresse MAC     |
----------------------------------------
| 255.255.255.255    | FF:FF:FF:FF:FF:FF |
----------------------------------------
```

Le **protocole ARP** (***Address Resolution Protocol***) fait la **correspondance entre les adresses MAC** (couche « Accès au Réseau » du modèle TCP/IP) **et les adresses IP** (couche « Internet »).

La commande **arp** permet de visualiser la table de correspondance entre ces adresses MAC et IP.

Ici les valeurs 255.255.255.255 et FF:FF:FF:FF:FF:FF indiquent que la table est vide.

10. Testez la liaison entre les deux ordinateurs en avec la commande `ping 192.168.1.2`. Les deux machines sont-elles bien connectées ?

Oui

11. Combien de paquets ont été envoyés par la commande ping pour tester la connexion entre les deux machines ??

4

La commande **`ping`** permet de **tester l'accessibilité d'une autre machine** à travers un réseau IP.

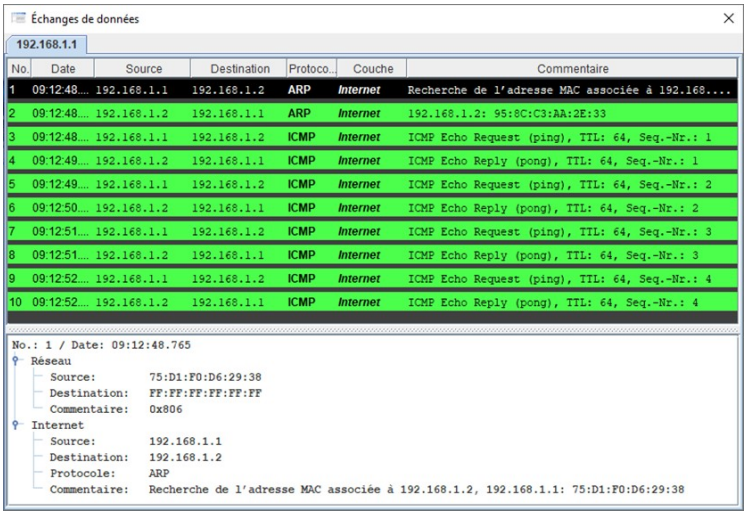
`ping` permet également de mesurer le temps mis pour recevoir une réponse. L'envoi est répété pour déterminer le taux de paquet perdu et le délai moyen.

12. Saisissez à nouveau la commande `arp` à l'invite de commande. Comment la table a été mise à jour ?

```
|     Adresse IP     |    Adresse MAC     |
----------------------------------------
| 192.168.1.2        | 95:8C:C3:AA:2E:33 |
| 255.255.255.255    | FF:FF:FF:FF:FF:FF |
----------------------------------------
```

La commande `ping` depuis l'ordinateur 192.168.1.1 vers 192.168.1.2 nécessite que 198.168.1.1 connaisse l'adresse MAC de 192.168.1.2. Or il ne la connaît pas. Un échange avec le protocole ARP lui a permis de l'obtenir et de mettre à jour sa table ARP.

13. Par un clic-droit sur la machine 192.168.1.1, affichez les échanges de données.



14. Cliquez sur les deux premières lignes de communication et observez dans la fenêtre du bas comment 192.168.1.1 a obtenu l'adresse MAC de 192.168.1.2 en utilisant le protocole ARP. En particulier à quelle adresse MAC avait-t-il envoyé la première requête ?

L'ordinateur 192.168.1.1 a envoyé une requête ARP à une destination inconnue (FF:FF:FF:FF:FF:FF).
L'ordinateur 192.168.1.2 lui a répondu en donnant son adresse MAC.

15. Observez les huit messages ICMP (*Internet Control Message Protocol*) suivants. Ils correspondent aux quatre paquets envoyés depuis 192.168.1.1 par la commande `ping` et aux quatre réponses envoyées par 192.168.1.2.

    Après chaque envoi d'un message « ICMP Echo Request (ping) » par 192.168.1.1, quelle est la réponse qu'il reçoit de 192.168.1.2 ?

    ICMP Echo Reply (pong)

    Observez dans la fenêtre du bas que seules les couches Réseau et Internet du modèle TCP/IP sont utilisées ici.

16. Quelles sont les adresses utilisées par la couche Réseau ?

    Les adresses MAC

17. A quelle couche OSI cela correspond-il ?

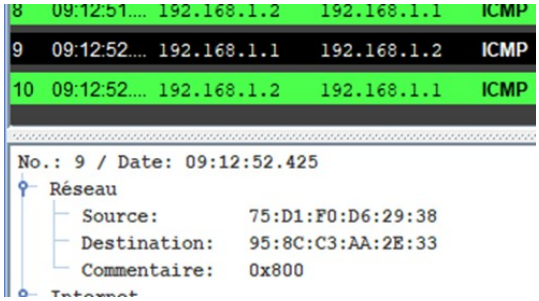
    Couches 1 (Physique) et 2 (Liaison de données)

18. Quelles sont les adresses utilisées par la couche Internet ?

    Les adresses IP

19. A quelle couche OSI cela correspond-il ?

    Couche 3 (Réseau)



20. Toujours en mode conception, ajoutez un ordinateur sur le même réseau avec le nom et adresse IP 192.168.1.10. Essayez de les relier aux autres par un câble. Que se passe-t-il ?

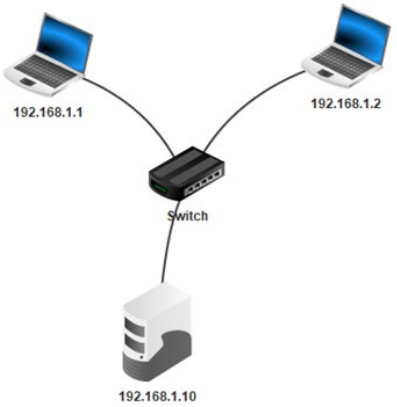
    Ce n'est pas possible, il n'y a plus de connecteur disponible.

On peut créer un réseau de plusieurs machines (ordinateurs, téléphones, réfrigérateurs connectés, voitures connectés, consoles de jeu,…) en les reliant entre elles physiquement par câble Ethernet ou Wi-Fi, etc. en utilisant un *hub* ou un *switch* :

- Un ***hub*** (ou **concentrateur**) est le matériel réseau le plus basique qui opère au niveau de la couche 1 (Physique) du modèle OSI. Il est utilisé pour un réseau local avec un nombre très limité de machines. Il n'est ni plus ni moins qu'une « multiprise RJ45 » qui envoie les données reçues sur tous ses ports.

- Un ***switch*** (ou **commutateur**) est un appareil "intelligent" qui opère au niveau de la couche 2 (Liaison de données) du modèle OSI. Il apprend l'adresse MAC de chaque appareil connecté et dirige les données uniquement vers le destinataire prévu.

21. Ajoutez un *switch* à votre réseau et câblez les 3 ordinateurs autour de lui pour former un réseau en **étoile**.



22. Passez en mode simulation ▶ et testez les connexions :
    - Depuis 192.168.1.1 vers 192.168.1.2. La connexion est-elle établie ?

      Oui
    - Depuis 192.168.1.1 vers 192.168.1.10. La connexion est-elle établie ?

      Oui

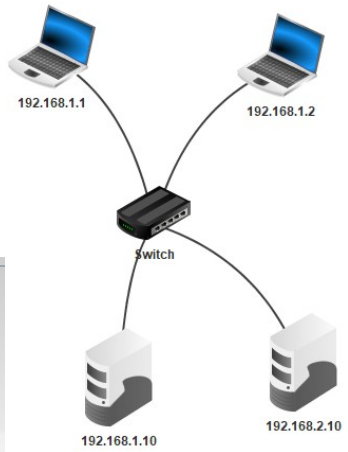
23. A l'aide de la commande arp, affichez la nouvelle table de correspondance entre adresses MAC et IP de l'ordinateur 192.168.1.1?

```
|     Adresse IP     |    Adresse MAC    |
----------------------------------------
| 192.168.1.2        | 95:8C:C3:AA:2E:33 |
| 192.168.1.10       | C9:EA:3B:52:C5:E5 |
| 255.255.255.255    | FF:FF:FF:FF:FF:FF |
----------------------------------------
```

Le **protocole IP** *(Internet Protocol*) est chargé d'acheminer au mieux les paquets de données d'une machine à une autre. Il permet d'identifier des machines grâce à leur adresse IP

Étudions maintenant de plus près les masques réseaux.

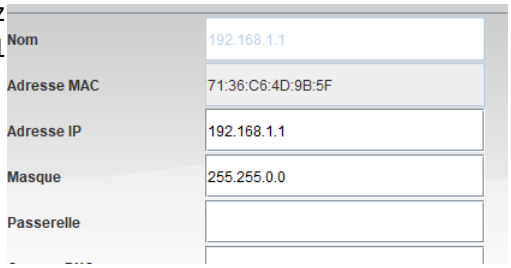
24. Revenez en mode conception et ajoutez un nouvel ordinateur avec le nom et l'adresse IP 192.168.2.10



25. Passez en mode simulation et testez la connexion depuis 192.168.1.1 vers 192.168.2.10. La connexion est-elle établie ?

Non

26. Passez en mode conception et changez les masques des deux ordinateurs 192.168.1.1 et 192.168.2.10 en 255.255.0.0.



27. Testez la connexion depuis 192.168.1.1 vers 192.168.2.10. La connexion est-elle établie maintenant?

Oui

Pour pouvoir communiquer, les deux ordinateurs doivent appartenir à un même réseau.

C'est le **masque** qui détermine **à quel réseau une adresse IP appartient** :

- Les **octets 255 du masque** désignent la partie de l'adresse IP qui **identifie réseau**.
- Les **octets 0 du masque** désignent la partie la partie de l'adresse IP qui **identifie une machine** sur le réseau.

```
                          Réseau →  ← machines
Adresse IP :        192. 168. 1 . 1
Masque :            255. 255.255. 0
Adresse réseau :    192. 168. 1 . 0
```

Avec un masque de 255.255.255.0, les ordinateurs qui ont des adresses IP qui commencent par les 3 mêmes premiers octets appartiennent au même réseau :

- 192.168.1.1, 192.168.1.2 et 192.168.1.10 s'écrivent sous la forme par 192.168.1.xxx. Ils appartiennent au même réseau, nommé 192.168.1.0
- 192.168.2.10 ne commence pas par 192.168.1.xxx, il appartient à un réseau différent, nommé 192.168.2.0. Il ne peut donc pas communiquer avec les ordinateurs du premier réseau.

La **première adresse** de la plage d'adresse désigne **le réseau lui-même**; et la **dernière est l'adresse de diffusion** (ou *broadcast*) qui permet d'envoyer un message à toutes les machines sur ce réseau, elles ne sont pas utilisées pour une machine. On a donc 2^8 - 2 = 254 adresses machines disponibles pour toutes les machines sur ce réseau.

```
192.168.1.0 (adresse du réseau)
192.168.1.1
192.168.1.2
…
192.168.1.254
192.168.1.255 (broadcast)
```

Par contre lorsqu'on change le masque de 192.168.1.1 et 192.168.2.10 en 255.255.0.0, les deux ordinateurs dont les adresses commencent par les mêmes 2 premiers octets appartiennent à un même réseau, nommé 192.168.0.0[3]. Ils peuvent alors communiquer entre eux directement.

```
                          Réseau → ← machines
Adresse IP :         192. 168. 1 . 1
Masque :             255. 255. 0 . 0
Adresse réseau :     192. 168. 0 . 0
```

Une autre notation souvent utilisée est la **notation CIDR** (*Classless Inter-Domain Routing*) qui consiste à **noter directement le nombre de bits significatifs en décimal**. L'adresse IP 192.168.1.1/255.255.255.0 peut écrire 192.168.1.1/24, pour indiquer que la partie réseau de l'adresse est de 24 bits.

28. Supprimez l'ordinateur 192.168.2.10 et remettez la masque de 192.168.1.1 à 255.255.255.0.

3 Les 65 534 (= 2^{16} - 2) adresses comprises entre 192.168.0.0 (réseau) et 192.168.255.255 (*broadcast*) sont sur ce même réseau.

Couche « Transport »

29. On va maintenant configurer l'ordinateur 192.168.1.10 en serveur. Par un double-clic sur cet ordinateur, ouvrez l'installateur de logiciels et installez l'application Serveur générique.
30. Ouvrez cette application et cliquez sur Démarrer le serveur (Port 55555).
31. De la même façon, ouvrez par un double-clic sur 192.168.1.1 l'installateur de logiciels et installez l'application Client générique. Ouvrez cette application et cliquez sur Connecter à l'adresse du serveur 192.168.1.10 sur le port 55555.

    Quand plusieurs programmes fonctionnent en même temps sur le même ordinateur (par exemple un navigateur, un logiciel d'email, etc.) il est nécessaire de savoir auquel est destiné chaque paquet IP.

    Un **port** logiciel[4] est un **numéro unique** (codé sur 16 bit[5]). Il y a donc 65536 ports différents possibles mais dont certains sont réservés.) associé à chaque programme sur un même ordinateur qui permet de savoir à quel programme sont destinés les paquets IP.

32. Affichez les échanges de données de la machine 192.168.1.1.

    Les lignes vertes correspondent aux échanges au niveau de la couche Internet (modèle TCP/IP). A quelle couche correspondent les lignes bleues les plus récents (faites défiler la liste vers le bas) ?

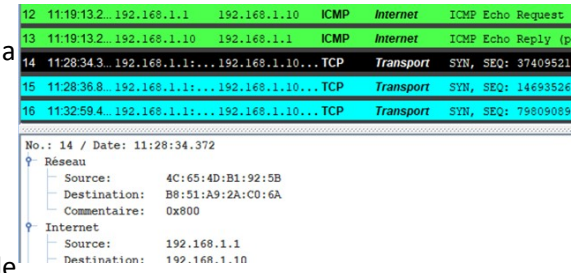
    Couche Transport

33. Sélectionnez une ligne bleue. Quel est le protocole utilisé par la couche Internet?

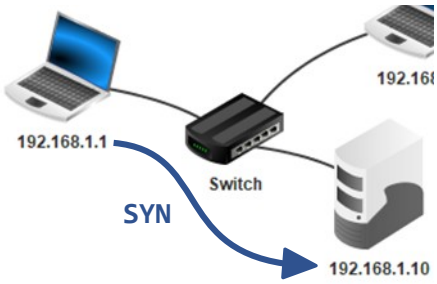
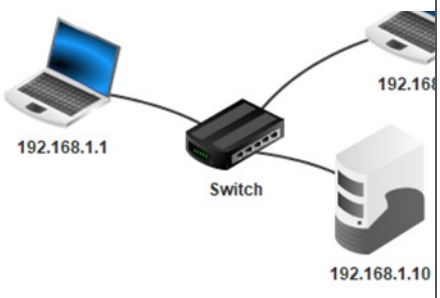
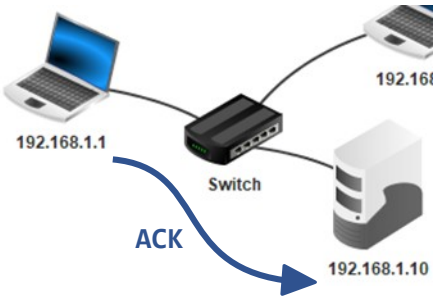
    IP

34. Quel est le protocole utilisé par la couche Transport?

    TCP



L'ouverture de la connexion TCP entre le client (192.168.1.1) et le serveur (192.168.1.10) se fait en trois étapes, appelées *three-way handshake*, qui correspondent à l'échange de trois paquets SYN, SYN-ACK et ACK[6] sont échangés. Observez les échanges entre le client et le serveur et renseignez les numéros de séquences échangés.

1. **SYN** (synchronized)	2.**SYN-ACK** (synchronize, acknowledge)	3.**ACK** (acknowledge)
Le client envoie un paquet SYN au serveur de demande de connexion Client→Serveur. 	Le serveur renvoie un paquet SYN-ACK d'acceptation de la connexion Client→Serveur et demande d'une nouvelle connexion Serveur→Client. 	Le client renvoie au serveur un paquet ACK d'acceptation de la connexion Serveur→Client. 
SEQ : 1195632247	ACK : 1195632248 / SEQ : 1605811125	ACK : 1605811126

Une fois le t*hree-way handshake* effectué, la communication full-duplex est établie entre le client et le serveur.

4 Il existe aussi des ports matériels (USB, HDMI, etc.) sur lesquels on connecte un appareil.
5 Il y a donc 65536 ports différents possibles mais dont certains sont réservés.
6 L'attaque par déni de service (DDoS) consiste à rendre un serveur web indisponible en lui un envoyant un très grand nombre de demandes de connexion (SYN Flood) jusqu'à atteindre la limite de SYN-ACK en attente d'une réponse envoyés par le serveur.

35. Pour demander la connexion au serveur, le client envoie un paquet SYN avec un numéro de séquence (SEQ) qui prend une valeur aléatoire, appelée ISN (Initial Sequence Number). Que peut-on dire du numéro ACK renvoyé par le serveur dans le paquet SYN-ACK?

    Le numéro du ACK est égal au numéro de séquence du paquet précédent (SYN) incrémenté de un.

36. Pour demander la connexion au client, le serveur envoie un paquet SYN-ACK avec un autre numéro de séquence (SEQ) aléatoire. Que peut-on dire du numéro ACK renvoyé par le client dans le paquet ACK ?

    Le numéro du ACK est égal au numéro de séquence du paquet précédent (SYN-ACK) incrémenté de un.

37. Quel est l'intérêt chaque fois de renvoyer les numéros de séquence incrémenté de un ?

    Confirmer à l'envoyeur que le message qu'il a envoyé a bien été reçu.

Le protocole **TCP** (***Transmission Control Protocol***), permet d'**établir une connexion** entre deux machines afin d'échanger des données. Les données sont **découpées en « petits » paquets** pour ne pas encombrer les réseaux, et éviter de renvoyer la totalité des données quand il y a une erreur.

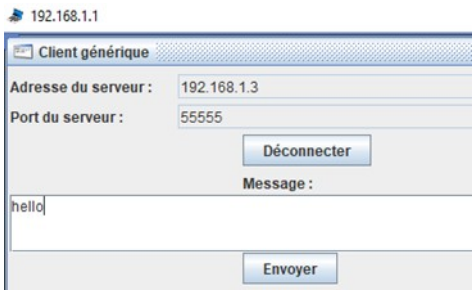
Pour ce faire, TCP joue plusieurs rôles : **découper des données en paquets**, **numéroter** les paquets ; **envoyer** les paquets les uns après les autres ; **vérifier** que tous les paquets sont arrivés ; **redemander** les paquets manquants, **remettre** les paquets **dans l'ordre**.

Le protocole TCP (Transmission Control Protocol) est un **protocole de communication fiable** entre applications. Ce n'est pas le seul protocole de la couche Transport. Il existe aussi le protocole **UDP** (*User Datagram Protocol*) qui transmet les paquets "sans connexion" (paquets non numérotés, sans renvoi si un paquet est perdu ou corrompu en chemin, etc. Plus rapide, UDP est surtout utilisé dans les cas où la vitesse prime, par exemple le streaming vidéo.

Couche « Application»

38. Retournez sur la fenêtre du Client générique de la machine 192.168.1.1 et entrez le message « hello » dans la fenêtre Message et Envoyer. Quelle est la réponse renvoyée par le serveur ?

    ```
    <<hello
    >>hello
    ```



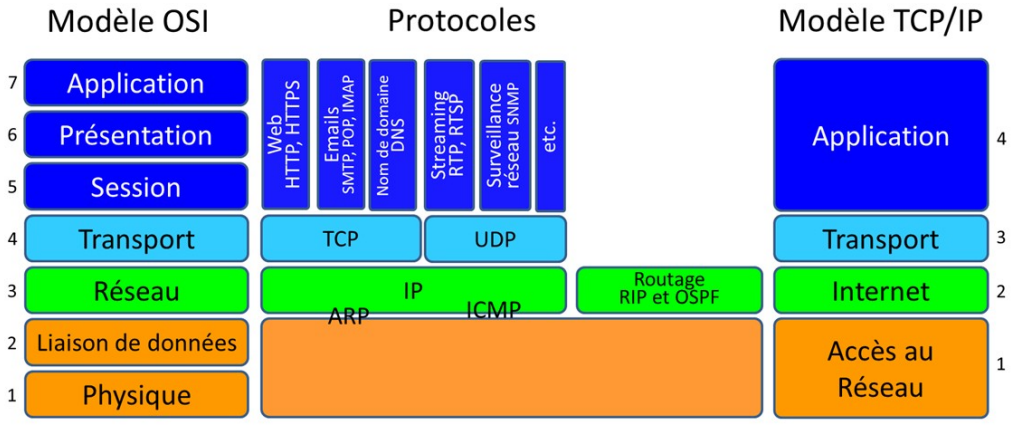
39. Retournez sur l'affichage des échanges de données de 192.168.1.1

    Application

40. Revenez en mode conception  et enregistrez votre travail.

Pile de protocoles

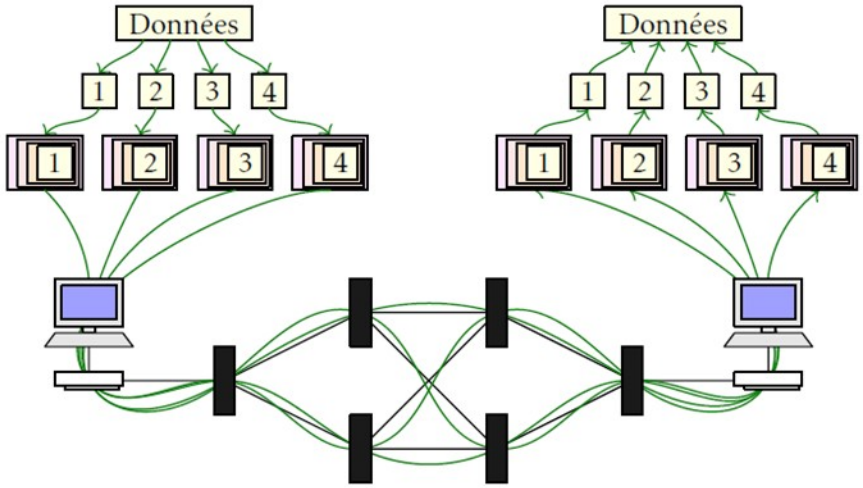
De nombreuses applications sur internet existent au niveau de la couche application, chacune utilisant des protocoles différents. En voici quelques exemple parmi bien d'autres: le Web avec protocoles HTTP et HTTPS, les emails avec SMTP, POP et IMAP, la visio et le streaming avec RTP, etc.



Pour s'échanger des données sur un réseau, chaque protocole s'appuie sur un protocole de la couche inférieure et lui rajoute des fonctionnalités.

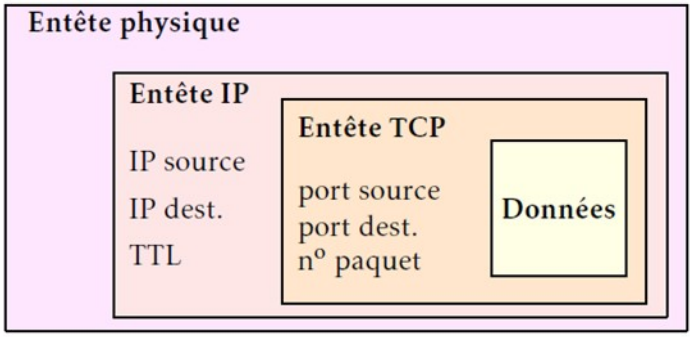
Découpage des données en paquets et encapsulation

Lors d'une transmission, les données traversent chacune des couches au niveau de la machine émettrice. À chaque couche, une information est ajoutée au paquet de données, il s'agit d'un entête, ensemble d'informations qui garantit la transmission. Au niveau de la machine réceptrice, lors du passage dans chaque couche, l'en-tête est lu, puis supprimé. Ainsi à la réception, le message est dans son état originel.



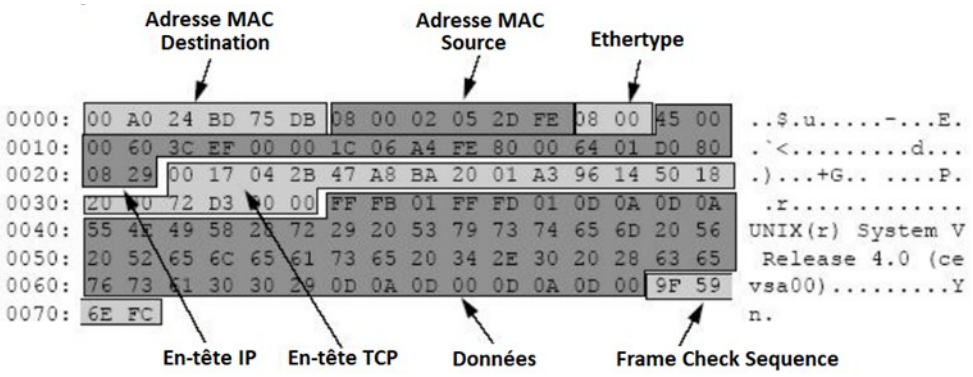
Chaque protocole rajoute des informations supplémentaires aux informations qu'il transmet. C'est l'**encapsulation**.

- un fragment de donnée est encapsulé dans un paquet TCP qui ajoute un no de paquet et les ports d'origine et de destination pour identifier l'application ;
- ce paquet est lui-même encapsulé dans un paquet IP, qui ajoute les adresses IP d'origine et du destinataire, ainsi qu'un compteur TTL (Time To Live) ;
- ce dernier étant alors envoyé via un protocole de la couche de liaison (par exemple Ethernet ou 4G) qui ajoute des informations supplémentaires.



Même si on utilise souvent le terme « paquet » pour toutes les couches du modèle, il change d'aspect ainsi l'appellation change aussi suivant les couches. On parle de message au niveau de la couche application ; de segment au niveau de la couche transport ; de paquet au niveau de la couche réseau ; de trame au niveau de la couche liaison ; et de signal au niveau de la couche physique.

Un exemple de trame Ethernet :

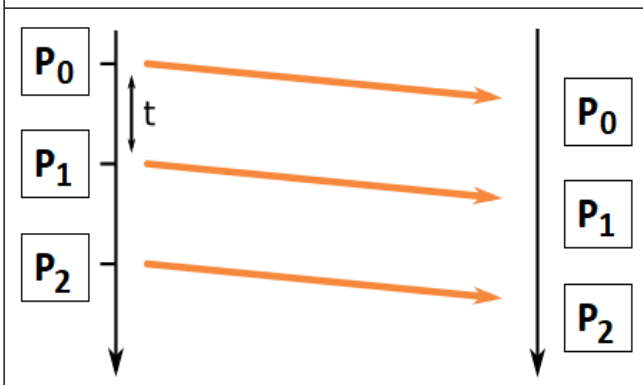
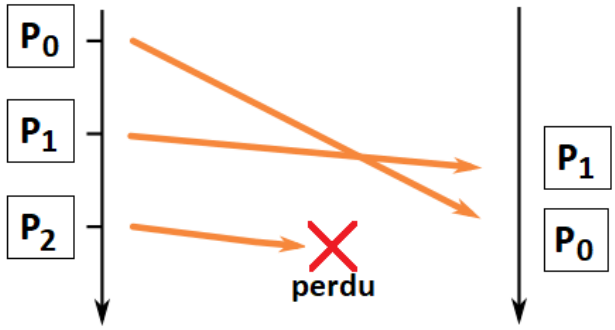


Protocole du bit alterné

Les données en sont pas envoyées en un seul bloc sur un réseau, mais découpées en « petits » paquets. Mais comment savoir si tous les paquets sont bien arrivés dans le bon ordre ou que certains paquets perdus doivent être renvoyés ? **Le protocole TCP numérote les paquets afin de les remettre dans l'ordre, et de redemander spécifiquement l'envoi des paquets perdus**. C'est très efficace, mais cher en ressources.

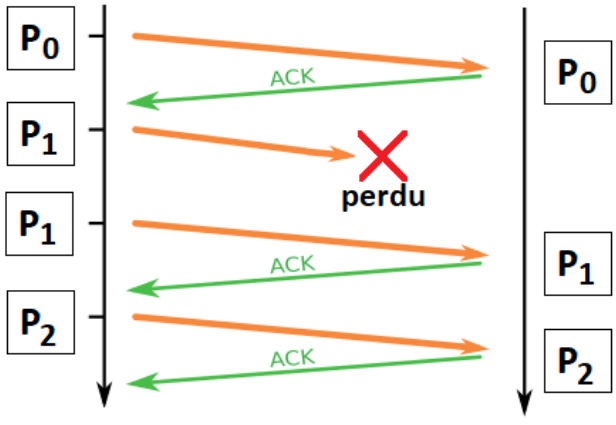
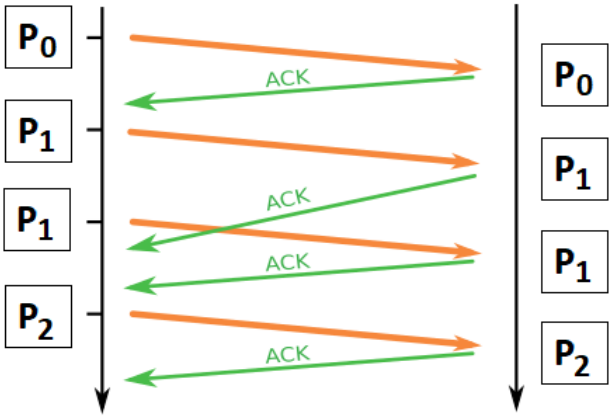
On étudie ici une approche simplifiée : le **protocole du bit alterné**.

On cherche à échanger des données découpées en trois paquets P_0, P_1 et P_2 envoyés à une cadence t fixe.

Transmission correcte	**Transmission incorrecte** P_0 est arrivé après P_1, P_2 est perdu.
	
Paquets reçus : P_0 P_1 P_2	Paquets reçus : P_1 P_0

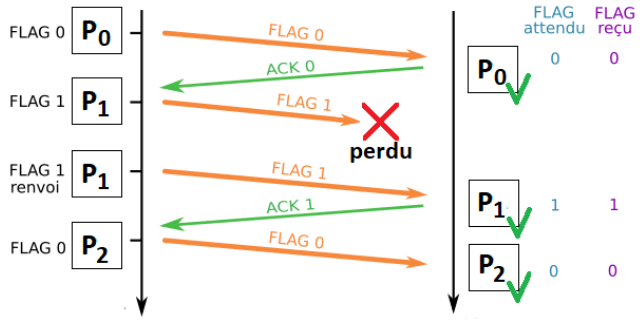
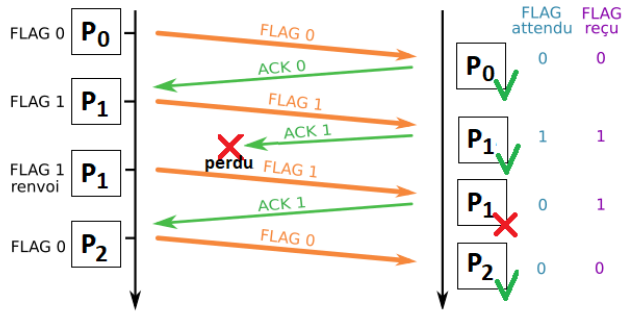
Une solution naïve consiste à renvoyer un accusé de réception ACK (*acknowledge*) pour confirmer qu'un paquet est bien arrivé. Si on ne reçoit pas de message ACK, on renvoie le paquet.

41. Complétez le schéma ci-dessous en indiquant les paquets reçus dans les deux cas.

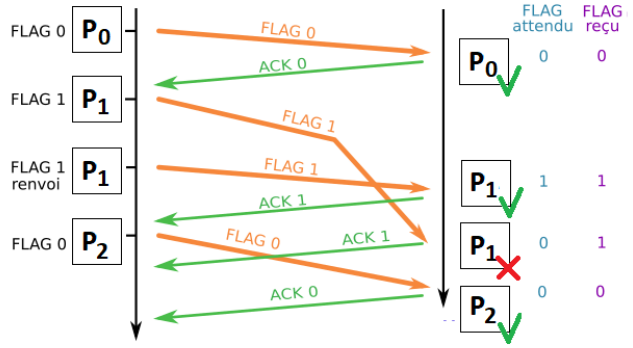
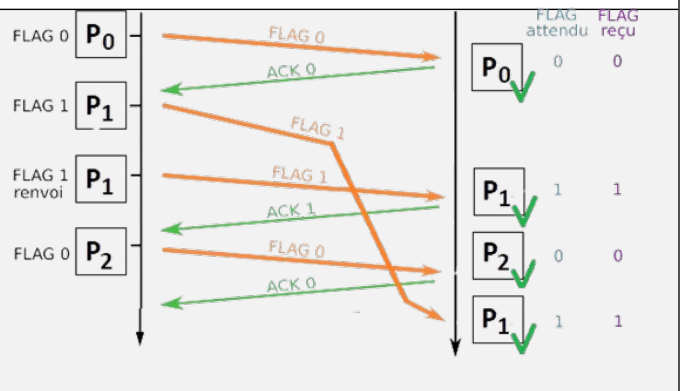
Transmission correcte Le ACK de P_1 n'a pas été reçu, P_1 est renvoyé.	**Transmission incorrecte** Le ACK de P_1 arrive trop tard, P_1 est envoyé deux fois.
	
Paquets reçus : P_0 P_1 P_2	Paquets reçus : P_0 P_1 P_1 P_2

Le **protocole du bit alterné** consiste à rajouter à chacun des paquets **un bit de contrôle, appelé FLAG** (drapeau en français) : le premier paquet est envoyé avec un FLAG à 0, dès qu'il est reçu, un accusé de réception ACK est renvoyé avec le FLAG reçu. Si on reçoit le message ACK avec le FLAG attendu, on peut envoyer le paquet suivant avec le FLAG à 1, sinon on renvoie le paquet précédent, et ainsi de suite…

Le protocole du bit alterné est très efficace dans les cas où un paquet ou un message ACK se perd : le paquet est renvoyé :

Cas du message perdu Le ACK de P_1 n'est pas reçu, P_1 est renvoyé	**Cas du ACK perdu** Le FLAG permet de détecter le doublon de P_1
	
Paquets reçus : P_0 P_1 P_2	Paquets reçus : P_0 P_1 P_2

Le FLAG permet aussi de gérer un paquet en retard :

Cas où un message est en retard Le FLAG permet de détecter le doublon de P_1.	**Transmission incorrecte** P_1 arrive après P_2
	
Paquets reçus : P_0 P_1 P_2	Paquets reçus : P_0 P_1 P_2 P_1

… mais pas dans tous cas !

42. Compléter le schéma ci-dessus du cas ou un paquet P_1 en retard arrive après P_2.