UTILITAIRES ET SERVICES TCP/IP, PROTOCOLES TCP/IP, LE ROUTAGE, LES ADRESSE IP, …

1) Liste des utilitaires et services TCP/IP :

Commande	Descriptif
arp	Affiche et modifie les entrées du cache ARP (Address Resolution Protocol), qui contient une ou plusieurs tables permettant de stocker les adresses IP et leurs adresses physiques Ethernet ou Token Ring résolues. À chaque carte réseau Ethernet ou Token Ring installée sur l'ordinateur correspond une table distincte. Utilisée sans paramètres, la commande **arp** affiche de l'aide.
finger	Affiche des informations sur un ou plusieurs utilisateurs sur un ordinateur distant spécifié (généralement un ordinateur sous UNIX) qui exécute le service Finger ou démon. L'ordinateur distant spécifie le format et la sortie de l'affichage des informations de l'utilisateur. Utilisé sans paramètres, **finger** affiche des informations d'aide.
ftp	Transfère des fichiers vers et depuis un ordinateur exécutant un service de serveur FTP (File Transfer Protocol) comme les Services Internet (IIS, *Internet Information Services*). La commande **Ftp** peut être utilisée de façon interactive ou en mode par lot en traitant des fichiers texte ASCII.
hostname	Affiche le nom d'hôte inclus dans le nom complet de l'ordinateur.
ipconfig	Affiche toutes les valeurs actuelles de la configuration du réseau TCP/IP et actualise les paramètres DHCP (Dynamic Host Configuration Protocol) et DNS (Domain Name System). Utilisé sans paramètres, **ipconfig** affiche l'adresse IP, le masque de sous-réseau et la passerelle par défaut de toutes les cartes.
lpq	Affiche l'état d'une file d'attente d'impression sur un ordinateur exécutant LPD (Line Printer Daemon). Sans paramètres, **lpq** affiche l'aide de la ligne de commande associée à la commande **lpq**.
lpr	Envoie un fichier vers un ordinateur exécutant LPD (Line Printer Daemon) en vue de son impression. Sans paramètres, **lpr** affiche l'aide de la ligne de commande associée à la commande **lpr**.
nbtstat	Affiche les statistiques du protocole NetBIOS sur TCP/IP (NetBT), les tables de noms NetBIOS associées à l'ordinateur local et aux ordinateurs distants ainsi que le cache de noms NetBIOS. **Nbtstat** permet d'actualiser le cache de noms NetBIOS et les noms inscrits avec le service de nom Internet Windows (WINS). Utilisée sans paramètres, la commande **nbtstat** affiche l'aide.
net send	Envoie des messages à d'autres utilisateurs, ordinateurs ou noms de messagerie sur le réseau.
net start	Démarre un service. Utilisée sans paramètre, la commande **net start** affiche la liste des services en cours d'exécution.
netstat	Affiche les connexions TCP actives, les ports sur lesquels l'ordinateur procède à l'écoute, la table de routage IP ainsi que des statistiques Ethernet, IPv4 (pour les protocoles IP, ICMP, TCP et UDP) et IPv6 (pour les protocoles IPv6, ICMPv6, TCP sur IPv6 et UDP sur IPv6). Utilisée sans paramètre, la commande **netstat** affiche les connexions TCP actives.
net stop	Arrête un service en cours d'exécution.
net use	Connecte ou déconnecte un ordinateur d'une ressource partagée, ou affiche des informations relatives aux connexions de l'ordinateur. Cette commande contrôle aussi les connexions réseau persistantes. Utilisée sans paramètre, la commande **net use** extrait une liste des connexions réseau.
net view	Affiche la liste des domaines, des ordinateurs ou des ressources partagées par l'ordinateur spécifié. Utilisée sans paramètre, la commande **net view** affiche la liste des ordinateurs de votre domaine en cours.

Commande	Descriptif
nslookup	Affiche des informations que vous pouvez utiliser pour diagnostiquer l'infrastructure DNS (Domain Name System). Avant d'utiliser cet outil, vous devez vous familiariser avec le fonctionnement du DNS. L'outil de la ligne de commande Nslookup est disponible uniquement si vous avez installé le protocole TCP/IP.
ping	Vérifie la connectivité IP à un autre ordinateur TCP/IP en envoyant des messages Requête d'écho ICMP (Internet Control Message Protocol). Le reçu des messages Réponse à écho correspondants s'affiche, ainsi que les temps des parcours circulaires. Ping est la principale commande TCP/IP utilisée pour résoudre les problèmes de connectivité, d'accessibilité et de résolution de nom. Utilisée sans paramètres, la commande **ping** affiche l'aide.
rcp	Permet de copier des fichiers entre un ordinateur fonctionnant sous Windows XP et un système qui exécute le service **rshd**, le service du noyau distant (le démon). Windows XP et Windows 2000 ne fournissent pas le service rshd. Utilisée sans paramètres, la commande **rcp** permet d'afficher l'aide.
rexec	Exécute des commandes sur des ordinateurs distants utilisant le service Rexec (démon). La commande **rexec** authentifie le nom d'utilisateur sur l'ordinateur distant avant d'exécuter la commande spécifiée. Windows XP et Windows 2000 ne fournissent pas le service Rexec. Utilisée sans paramètres, la commande **rexec** permet d'afficher l'aide.
route	Affiche et modifie les entrées dans la table de routage IP locale. Utilisée sans paramètres, la commande **route** permet d'afficher l'aide.
rsh	Permet d'exécuter des commandes sur des ordinateurs distants qui exécutent le service ou le démon RSH. Windows XP et Windows 2000 ne fournissent pas de service RSH. Un service RSH (Rshsvc.exe) est fourni avec le kit de ressources techniques Windows 2000. Utilisée sans paramètres, la commande **rsh** permet d'afficher l'aide.
telnet	Les commandes Telnet écrites dans le contexte Telnet vous permettent de communiquer avec un ordinateur distant utilisant le protocole Telnet. L'exécution de Telnet sans paramètres vous permet d'entrer dans le contexte Telnet. Une fois au niveau de l'invite de commande, vous pouvez utiliser les commandes Telnet pour gérer un ordinateur exécutant le client Telnet.
tftp	Permet le transfert de fichiers depuis et vers un ordinateur distant, qui en général fonctionne sous UNIX et exécute le service ou le démon TFTP (Trivial File Transfer Protocol). Utilisée sans paramètres, la commande **tftp** permet d'afficher l'aide.
tracert	Détermine l'itinéraire menant vers une destination par la transmission de messages ICMP (messages Requête d'écho Internet Control Message Protocol) en augmentant de façon incrémentielle les valeurs des champs TTL (Time to Live, Durée de vie). L'itinéraire affiché correspond à la série d'interfaces de routeurs les plus proches des routeurs sur l'itinéraire situé entre un hôte source et une destination. L'interface la plus proche est celle du routeur qui est la plus proche de l'hôte émetteur sur l'itinéraire. Utilisée sans paramètres, la commande **tracert** permet d'afficher l'aide.
winipcfg	Permet, aux utilisateurs ou aux administrateurs, de configuration IP et d'afficher l'adresse IP en cours ainsi que d'autres informations utiles concernant la configuration réseau. Vous pouvez réinitialiser une ou plusieurs adresses IP. Les boutons Libérer ou Renouveler libèrent ou renouvellent respectivement une adresse IP. Si vous souhaitez libérer ou renouveler toutes les adresses IP, cliquez sur Tout libérer ou Tout renouveler. Lorsque l'un de ces boutons est activé, une nouvelle adresse IP est obtenue soit à partir du service DHCP, soit à partir de l'ordinateur qui s'affecte lui-même une adresse IP privée automatique.

2) Les différents protocoles TCP/IP :

2-1) Que signifie TCP/IP ?

TCP/IP est une suite de protocoles. Le sigle TCP/IP signifie «**Transmission Control Protocol/Internet Protocol**» et se prononce «T-C-P-I-P». Il provient des noms des deux protocoles majeurs de la suite de protocoles, c'est-à-dire les protocoles TCP et IP).

TCP/IP représente d'une certaine façon l'ensemble des règles de communication sur internet et se base sur la notion adressage IP, c'est-à-dire le fait de fournir une adresse IP à chaque machine du réseau afin de pouvoir acheminer des paquets de données. Etant donné que la suite de protocoles TCP/IP a été créée à l'origine dans un but militaire, elle est conçue pour répondre à un certain nombre de critères parmi lesquels :

➔ Le fractionnement des messages en paquets;

➔ L'utilisation d'un système d'adresses;

➔ L'acheminement des données sur le réseau (routage);

➔ Le contrôle des erreurs de transmission de données.

La connaissance de l'ensemble des protocoles TCP/IP n'est pas essentielle pour un simple utilisateur, au même titre qu'un téléspectateur n'a pas besoin de connaître le fonctionnement de son téléviseur, ni des réseaux audiovisuels. Toutefois, sa connaissance est nécessaire pour les personnes désirant administrer ou maintenir un réseau TCP/IP.

2-2) Différence entre standard et implémentation

TCP/IP regroupe globalement deux notions :

➔ La notion de **standard** : TCP/IP représente la façon dont les communications s'effectuent sur un réseau.

➔ La notion d'**implémentation** : l'appellation TCP/IP est souvent étendue aux logiciels basés sur le protocole TCP/IP. TCP/IP est en fait un modèle sur lequel les développeurs d'applications réseau s'appuient. Les applications sont ainsi des implémentations du protocole TCP/IP.

2-3) TCP/IP est un modèle en couches

Afin de pouvoir appliquer le modèle TCP/IP à n'importe quelles machines, c'est-à-dire indépendamment du système d'exploitation, le système de protocoles TCP/IP a été décomposé en plusieurs modules effectuant chacun une tâche précise. De plus, ces modules effectuent ces tâches les uns après les autres dans un ordre précis, on a donc un système stratifié, c'est la raison pour laquelle on parle de **modèle en couches**.

Le terme de couche est utilisé pour évoquer le fait que les données qui transitent sur le réseau traversent plusieurs **niveaux de protocoles**. Ainsi, les données (paquets d'informations) qui circulent sur le réseau sont traitées successivement par chaque couche, qui vient rajouter un élément d'information (appelé *en-tête*) puis sont transmises à la couche suivante.

Le modèle TCP/IP est très proche du modèle OSI (modèle comportant 7 couches) qui a été mis au point par l'organisation internationale des standards (ISO, *organisation internationale de normalisation*) afin de normaliser les communications entre ordinateurs.

2-4) Présentation du modèle OSI

OSI signifie *Open Systems Interconnection*, ce qui se traduit par *Interconnexion de systèmes ouverts*. Ce modèle a été mis en place par l'ISO afin de mettre en place un standard de communications entre les ordinateurs d'un réseau, c'est-à-dire les règles qui gèrent les communications entre des ordinateurs. En effet, aux origines des réseaux chaque constructeur avait un système propre (on parle de système propriétaire). Ainsi de nombreux réseaux incompatibles coexistaient. C'est la raison pour laquelle l'établissement d'une norme a été nécessaire.

Le rôle du modèle OSI consiste à standardiser la communication entre les machines afin que différents constructeurs puissent mettre au point des produits (logiciels ou matériels) compatibles (pour peu qu'ils respectent scrupuleusement le modèle OSI).

2-5) L'intérêt d'un système en couches

Le but d'un système en couches est de séparer le problème en différentes parties (les couches) selon leur niveau d'abstraction.

Chaque couche du modèle communique avec une couche adjacente (celle du dessus ou celle du dessous). Chaque couche utilise ainsi les services des couches inférieures et en fournit à celle de niveau supérieur.

2-6) Le modèle OSI

le modèle OSI est un modèle qui comporte 7 couches, tandis que le modèle TCP/IP n'en comporte que 4. En réalité le modèle TCP/IP a été développé à peu près au même moment que le modèle OSI, c'est la raison pour laquelle il s'en inspire mais n'est pas totalement conforme aux spécifications du modèle OSI. Les couches du modèle OSI sont les suivantes :

Niveau	Ancien modèle	Nouveau modèle
Niveau 7	Couche Application	Niveau Application
Niveau 6	Couche Présentation	Niveau Présentation
Niveau 5	Couche Session	Niveau Session
Niveau 4	Couche Transport	Niveau Message
Niveau 3	Couche Réseau	Niveau Paquet
Niveau 2	Couche Liaison données	Niveau Trame
Niveau 1	Couche Physique	Niveau Physique

➔ **La couche physique** définit la façon dont les données sont physiquement converties en signaux numériques sur le média de communication (impulsions électriques, modulation de la lumière, etc.).

➔ **La couche liaison données** définit l'interface avec la carte réseau et le partage du média de transmission.

➔ **La couche réseau** permet de gérer l'adressage et le routage des données, c'est-à-dire leur acheminement via le réseau.

➔ **La couche transport** est chargée du transport des données, de leur découpage en paquets et de la gestion des éventuelles erreurs de transmission.

➔ **La couche session** définit l'ouverture et la destruction des sessions de communication entre les machines du réseau.

➔ **La couche présentation** définit le format des données manipulées par le niveau applicatif (leur représentation, éventuellement leur compression et leur chiffrement) indépendamment du système.

➔ **La couche application** assure l'interface avec les applications. Il s'agit donc du niveau le plus proche des utilisateurs, géré directement par les logiciels.

2-7) Le modèle TCP/IP

Le modèle TCP/IP, inspiré du modèle OSI, reprend l'approche modulaire (utilisation de modules ou couches) mais en contient uniquement quatre :

Modèle TCP/IP	Modèle OSI
Couche Application	Couche Application
	Couche Présentation
	Couche Session
Couche Transport (TCP)	Couche Transport
Couche Internet (IP)	Couche Réseau
Couche Accès réseau	Couche Liaison données
	Couche Physique

Comme on peut le remarquer, les couches du modèle TCP/IP ont des tâches beaucoup plus diverses que les couches du modèle OSI, étant donné que certaines couches du modèle TCP/IP correspondent à plusieurs couches du modèle OSI.

Les rôles des différentes couches sont les suivants :

➔ **Couche Accès réseau** : elle spécifie la forme sous laquelle les données doivent être acheminées quel que soit le type de réseau utilisé

➔ **Couche Internet** : elle est chargée de fournir le paquet de données (datagramme)

➔ **Couche Transport** : elle assure l'acheminement des données, ainsi que les mécanismes permettant de connaître l'état de la transmission

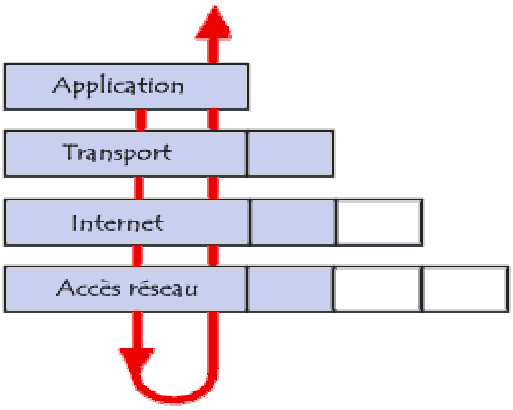
➔ **Couche Application** : elle englobe les applications standard du réseau (Telnet, SMTP, FTP, ...)

Voici les principaux protocoles faisant partie de la suite TCP/IP :

Modèle TCP/IP
Couche Application Applications réseau
Couche Transport TCP ou UDP
Couche Internet IP, ARP, RARP
Couche Accès réseau FTS, FDDI, PPP, Ethernet, Anneau à jeton (Token ring)
Couche Physique

2-8) Encapsulation des données

Lors d'une transmission, les données traversent chacune des couches au niveau de la machine émettrice. A chaque couche, une information est ajoutée au paquet de données, il s'agit d'un **en-tête**, 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...



A chaque niveau, le paquet de données change d'aspect, car on lui ajoute un en-tête, ainsi les appellations changent suivant les couches :

➔ Le paquet de données est appelé **message** au niveau de la couche Application

➔ Le message est ensuite encapsulé sous forme de **segment** dans la couche Transport

➔ Le segment une fois encapsulé dans la couche Internet prend le nom de **datagramme**

➔ Enfin, on parle de **trame** au niveau de la couche Accès réseau

2-9) La couche Accès réseau

La couche accès réseau est la première couche de la pile TCP/IP, elle offre les capacités à accéder à un réseau physique quel qu'il soit, c'est-à-dire les moyens à mettre en oeuvre afin de transmettre des données via un réseau.

Ainsi, la couche accès réseau contient toutes les spécifications concernant la transmission de données sur un réseau physique, qu'il s'agisse de réseau local (Anneau à jeton - token ring, ethernet, FDDI), de connexion à une ligne téléphonique ou n'importe quel type de liaison à un réseau. Elle prend en charge les notions suivantes :

➔ Acheminement des données sur la liaison

➔ Coordination de la transmission de données (synchronisation)

➔ Format des données

➔ Conversion des signaux (analogique/numérique)

➔ Contrôle des erreurs à l'arrivée

➔ ...

Heureusement toutes ces spécifications sont transparentes aux yeux de l'utilisateur, car l'ensemble de ces tâches est en fait réalisé par le système d'exploitation, ainsi que les drivers du matériel permettant la connexion au réseau (ex : driver de carte réseau).

2-10) La couche Internet

La couche Internet est la couche "la plus importante" (elles ont toutes leur importance) car c'est elle qui définit les datagrammes, et qui gère les notions d'adressage IP.

Elle permet l'acheminement des datagrammes (paquets de données) vers des machines distantes ainsi que de la gestion de leur fragmentation et de leur assemblage à réception.

La couche Internet contient 5 protocoles :

➔ Le protocole IP

➔ Le protocole ARP

➔ Le protocole ICMP

➔ Le protocole RARP

➔ Le protocole IGMP

Les trois premiers protocoles sont les protocoles les plus importants de cette couche...

2-11) La couche Transport

Les protocoles des couches précédentes permettaient d'envoyer des informations d'une machine à une autre. La couche transport permet à des applications tournant sur des machines distantes de communiquer. Le problème consiste à identifier ces applications.

En effet, suivant la machine et son système d'exploitation, l'application pourra être un programme, une tâche, un processus...

De plus, la dénomination de l'application peut varier d'un système à un autre, c'est la raison pour laquelle un système de numéro a été mis en place afin de pouvoir associer un type d'application à un type de données, ces identifiants sont appelés ports.

La couche transport contient deux protocoles permettant à deux applications d'échanger des données indépendamment du type de réseau emprunté (c'est-à-dire indépendamment des couches inférieures...), il s'agit des protocoles suivants :

- ➔ TCP, un protocole orienté connexion qui assure le contrôle des erreurs
- ➔ UDP, un protocole non orienté connexion dont le contrôle d'erreur est archaïque

2-12) La couche Application

La couche application est la couche située au sommet des couches de protocoles TCP/IP. Celle-ci contient les applications réseaux permettant de communiquer grâce aux couches inférieures.

Les logiciels de cette couche communiquent donc grâce à un des deux protocoles de la couche inférieure (la couche transport) c'est-à-dire TCP ou UDP.

Les applications de cette couche sont de différents types, mais la plupart sont des services réseau, c'est-à-dire des applications fournies à l'utilisateur pour assurer l'interface avec le système d'exploitation. On peut les classer selon les services qu'ils rendent :

- ➔ Les services de gestion (transfert) de fichier et d'impression
- ➔ Les services de connexion au réseau
- ➔ Les services de connexion à distance
- ➔ Les utilitaires Internet divers

3) Le protocole IP :

3-1) Le rôle du protocole IP

Le **protocole IP** fait partie de la couche Internet de la suite de protocoles TCP/IP. C'est un des protocoles les plus importants d'Internet car il permet l'élaboration et le transport des datagrammes IP (les paquets de données), sans toutefois en assurer la « livraison ». En réalité, le protocole IP traite les datagrammes IP indépendamment les uns des autres en définissant leur représentation, leur routage et leur expédition.

Le protocole IP détermine le destinataire du message grâce à 3 champs :

➔ Le champ adresse IP : adresse de la machine

➔ Le champ masque de sous-réseau : un masque de sous-réseau permet au protocole IP de déterminer la partie de l'adresse IP qui concerne le réseau

➔ Le champ passerelle par défaut : Permet au protocole Internet de savoir à quelle machine remettre le datagramme si jamais la machine de destination n'est pas sur le réseau local

3-2) Les datagrammes

Les données circulent sur Internet sous forme de datagrammes (on parle aussi de paquets). Les datagrammes sont des données encapsulées, c'est-à-dire des données auxquelles on a ajouté des en-têtes correspondant à des informations sur leur transport (telles que l'adresse IP de destination).

Les données contenues dans les datagrammes sont analysées (et éventuellement modifiées) par les routeurs permettant leur transit.

Voici ce à quoi ressemble un datagramme :

<table>
<tr><td><--</td><td colspan="3">32 bits</td><td>--></td></tr>
<tr><td>Version
(4 bits)</td><td>Longueur d'en-tête
(4 bits)</td><td>Type de service
(8 bits)</td><td colspan="2">Longueur totale
(16 bits)</td></tr>
<tr><td colspan="3">Identification
(16 bits)</td><td>Drapeau
(3 bits)</td><td>Décalage fragment
(13 bits)</td></tr>
<tr><td colspan="2">Durée de vie
(8 bits)</td><td>Protocole
(8 bits)</td><td colspan="2">Somme de contrôle en-tête
(16 bits)</td></tr>
<tr><td colspan="5">Adresse IP source (32 bits)</td></tr>
<tr><td colspan="5">Adresse IP destination (32 bits)</td></tr>
<tr><td colspan="5">Données</td></tr>
</table>

Voici la signification des différents champs :

➔ **Version** (4 bits) : il s'agit de la version du protocole IP que l'on utilise (actuellement on utilise la version 4 *IPv4*) afin de vérifier la validité du datagramme. Elle est codée sur 4 bits.

➔ **Longueur d'en-tête**, ou *IHL* pour *Internet Header Length* (4 bits) : il s'agit du nombre de mots de 32 bits constituant l'en-tête (nota : la valeur minimale est 5). Ce champ est codé sur 4 bits.

➔ **Type de service** (8 bits) : il indique la façon selon laquelle le datagramme doit être traité.

➔ **Longueur totale** (16 bits) : il indique la taille totale du datagramme en octets. La taille de ce champ étant de 2 octets, la taille totale du datagramme ne peut dépasser 65536 octets. Utilisé conjointement avec la taille de l'en-tête, ce champ permet de déterminer où sont situées les données.

➔ **Identification, drapeaux (flags) et déplacement de fragment** sont des champs qui permettent la fragmentation des datagrammes, ils sont expliqués plus bas.

➔ **Durée de vie** appelée aussi **TTL**, pour *Time To Live* (8 bits) : ce champ indique le nombre maximal de routeurs à travers lesquels le datagramme peut passer. Ainsi ce champ est décrémenté à chaque passage dans un routeur, lorsque celui-ci atteint la valeur critique de 0, le routeur détruit le datagramme. Cela évite l'encombrement du réseau par les datagrammes perdus.

➔ **Protocole** (8 bits) : ce champ, en notation décimale, permet de savoir de quel protocole est issu le datagramme

- ICMP : 1
- IGMP : 2
- TCP : 6
- UDP : 17

➔ **Somme de contrôle de l'en-tête, ou en anglais *header checksum* (16 bits)** : ce champ contient une valeur codée sur 16 bits qui permet de contrôler l'intégrité de l'en-tête afin de déterminer si celui-ci n'a pas été altéré pendant la transmission. La somme de contrôle est le complément à un de tous les mots de 16 bits de l'en-tête (champ *somme de contrôle* exclu). Celle-ci est en fait telle que lorsque l'on fait la somme des champs de l'en-tête (somme de contrôle incluse), on obtient un nombre avec tous les bits positionnés à 1

➔ **Adresse IP source** (32 bits) : Ce champ représente l'adresse IP de la machine émettrice, il permet au destinataire de répondre

➔ **Adresse IP destination** (32 bits) : adresse IP du destinataire du message

3-3) La fragmentation des datagrammes IP

Comme nous l'avons vu précédemment, la taille d'un datagramme maximale est de 65536 octets. Toutefois cette valeur n'est jamais atteinte car les réseaux n'ont pas une capacité suffisante pour envoyer de si gros paquets. De plus, les réseaux sur Internet utilisent différentes technologies, si bien que la taille maximale d'un datagramme varie suivant le type de réseau.

La taille maximale d'une trame est appelée *MTU* (Maximum Transfer Unit), elle entraînera la fragmentation du datagramme si celui-ci a une taille plus importante que le MTU du réseau.

Type de réseau	MTU (en octets)
Arpanet	1000
Ethernet	1500
FDDI	4470

La fragmentation d'un datagramme se fait au niveau des routeurs, c'est-à-dire lors de la transition d'un réseau dont le MTU est important à un réseau dont le MTU est plus faible. Si le datagramme est trop grand pour passer sur le réseau, le routeur va le fragmenter, c'est-à-dire le découper en fragments de tailles inférieures au MTU du réseau et de telle façon que la taille du fragment soit un multiple de 8 octets.

Le routeur va ensuite envoyer ces fragments de manière indépendante et les réencapsuler (ajouter un en-tête à chaque fragment) de telle façon à tenir compte de la nouvelle taille du fragment. De plus, le routeur ajoute des informations afin que la machine de destination puisse réassembler les fragments dans le bon ordre. Rien ne dit toutefois que les fragments arriveront dans le bon ordre, étant donné qu'ils sont acheminés indépendamment les uns des autres.

Pour tenir compte de la fragmentation, chaque datagramme possède plusieurs champs permettant leur réassemblage :

➔ **champ déplacement de fragment** (13 bits) : champ permettant de connaître la position du début du fragment dans le datagramme initial. L'unité de mesure de ce champ est de 8 octets (le premier fragment ayant une valeur de zéro).

➔ **champ identification** (16 bits) : numéro attribué à chaque fragment afin de permettre leur réassemblage.

➔ **champ longueur totale** (16 bits) : il est recalculé pour chaque fragment.

➔ **champ drapeau** (3 bits) : il est composé de trois bits :
- Le premier n'est pas utilisé.

- Le second (appelé **DF** : *Don't Fragment*) indique si le datagramme peut être fragmenté ou non. Si jamais un datagramme a ce bit positionné à un et que le routeur ne peut pas l'acheminer sans le fragmenter, alors le datagramme est rejeté avec un message d'erreur
- Le dernier (appelé **MF** : *More Fragments*, en français *Fragments à suivre*) indique si le datagramme est un fragment de donnée (1). Si l'indicateur est à zéro, cela indique que le fragment est le dernier (donc que le routeur devrait être en possession de tous les fragments précédents) ou bien que le datagramme n'a pas fait l'objet d'une fragmentation

3-4) Le routage IP

Le routage IP fait partie intégrante de la couche IP de la suite TCP/IP. Le routage consiste à assurer l'acheminement d'un datagramme IP à travers un réseau en empruntant le chemin le plus court. Ce rôle est assuré par des machines appelées routeurs, c'est-à-dire des machines reliées (reliant) au moins deux réseaux.

3-5) Plus d'informations

Pour plus d'informations le mieux est de se reporter à la RFC 791 expliquant de manière détaillée le protocole IP :

➔ RFC 791 traduite en français
➔ RFC 791 originale

4) Les adresses IP :

4-1) Qu'est-ce qu'une adresse IP ?

Sur Internet, les ordinateurs communiquent entre eux grâce au protocole IP (*Internet Protocol*), qui utilise des adresses numériques, appelées **adresses IP**, composées de 4 nombres entiers (4 octets) entre 0 et 255 et notées sous la forme xxx.xxx.xxx.xxx. Par exemple, *194.153.205.26* est une adresse IP donnée sous une forme technique.

Ces adresses servent aux ordinateurs du réseau pour communiquer entre-eux, ainsi chaque ordinateur d'un réseau possède une adresse IP unique sur ce réseau.

C'est l'ICANN (*Internet Corporation for Assigned Names and Numbers*, remplaçant l'IANA, *Internet Assigned Numbers Agency*, depuis 1998) qui est chargée d'attribuer des adresses IP publiques, c'est-à-dire les adresses IP des ordinateurs directement connectés sur le réseau public internet.

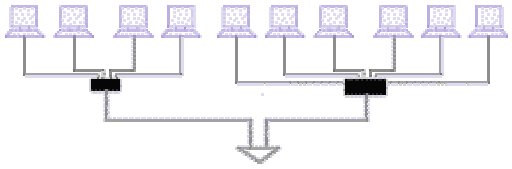
4-2) Déchiffrement d'une adresse IP

Une adresse IP est une adresse 32 bits, généralement notée sous forme de 4 nombres entiers séparés par des points. On distingue en fait deux parties dans l'adresse IP :

➔ Une partie des nombres à gauche désigne le réseau est est appelée **ID de réseau** (en anglais *netID*),

➔ Les nombres de droite désignent les ordinateurs de ce réseau est est appelée **ID d'hôte** (en anglais *host-ID*).

Soit l'exemple ci-dessous:



Notons le réseau de gauche *194.28.12.0*. Il contient les ordinateurs suivants :
➔ 194.28.12.1 à 194.28.12.4

Notons celui de droite *178.12.0.0*. Il comprend les ordinateurs suivants :
➔ 178.12.77.1 à 178.12.77.6

Dans le cas ci-dessus, les réseaux sont notés *194.28.12* et *178.12.77*, puis on numérote incrémentalement chacun des ordinateurs le constituant.

Imaginons un réseau noté *58.0.0.0*. Les ordinateurs de ce réseau pourront avoir les adresses IP allant de *58.0.0.1* à *58.255.255.254*. Il s'agit donc d'attribuer les numéros de telle façon qu'il y ait une organisation dans la hiérarchie des ordinateurs et des serveurs.

Ainsi, plus le nombre de bits réservé au réseau est petit, plus celui-ci peut contenir d'ordinateurs.

En effet, un réseau noté *102.0.0.0* peut contenir des ordinateurs dont l'adresse IP peut varier entre 102.0.0.1 et 102.255.255.254 (256*256*256-2=16777214 possibilités), tandis qu'un réseau noté *194.26* ne pourra contenir que des ordinateurs dont l'adresse IP sera comprise entre 194.26.0.1 et 194.26.255.254 (256*256-2=65534 possibilités), c'est la notion de **classe d'adresse IP**.

4-3) Adresses particulières

Lorsque l'on annule la partie host-id, c'est-à-dire lorsque l'on remplace les bits réservés aux machines du réseau par des zéros (par exemple *194.28.12.0*), on obtient ce que l'on appelle l'**adresse réseau**. Cette adresse ne peut être attribuée aucun des ordinateurs du réseau.

Lorsque la partie netid est annulée, c'est-à-dire lorsque les bits réservés au réseau sont remplacés par des zéros, on obtient l'**adresse machine**. Cette adresse représente la machine spécifiée par le host-ID qui se trouve sur le réseau courant.

Lorsque tous les bits de la partie host-id sont à 1, l'adresse obtenue est appellée l'**adresse de diffusion** (en anglais **broadcast**). Il s'agit d'une adresse spécifique, permettant d'envoyer un message à toutes les machines situées sur le réseau spécifié par le *netID*.

A l'inverse, lorsque tous les bits de la partie netid sont à 1, l'adresse obtenue constitue l'**adresse de diffusion limitée** (**multicast**).

Enfin, l'adresse **127.0.0.1** est appelée **adresse de rebouclage** (en anglais **loopback**), car elle désigne la **machine locale** (en anglais *localhost*).

4-4) Les classes de réseaux

Les adresses IP sont réparties en classes, selon le nombre d'octets qui représentent le réseau.

4-4-1) Classe A

Dans une adresse IP de classe A, le premier octet représente le réseau.

Le bit de poids fort (le premier bit, celui de gauche) est à zéro, ce qui signifie qu'il y a 2^7 (00000000 à 01111111) possibilités de réseaux, soit 128 possibilités. Toutefois, le réseau 0 (bits valant 00000000) n'existe pas et le nombre 127 est réservé pour désigner votre machine.

Les réseaux disponibles en classe A sont donc les réseaux allant de **1.0.0.0** à **126.0.0.0** (les derniers octets sont des zéros ce qui indique qu'il s'agit bien de réseaux et non d'ordinateurs !)

Les trois octets de droite représentent les ordinateurs du réseaux, le réseau peut donc contenir un nombre d'ordinateur égal à : 2^{24}-2 = 16777214 ordinateurs.

Une adresse IP de classe A, en binaire, ressemble à ceci :

0	xxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
Réseau		Ordinateurs		

Adressage IP de 1.xxx.xxx.xxx à 126.xxx.xxx.xxx

Masque de sous-réseau : 255.0.0.0

Elle contient 127 réseaux et 16 777 214 hôtes.

4-4-2) Classe B

Dans une adresse IP de classe B, les deux premiers octets représentent le réseau.

Les deux premiers bits sont 1 et 0, ce qui signifie qu'il y a 2^{14} (10 000000 00000000 à 10 111111 11111111) possibilités de réseaux, soit 16384 réseaux possibles.

Les réseaux disponibles en classe B sont donc les réseaux allant de **128.0.0.0** à **191.255.0.0**
Les deux octets de droite représentent les ordinateurs du réseau.

Le réseau peut donc contenir un nombre d'ordinateurs égal à : 2^{16}-2^1 = 65534 ordinateurs.

Une adresse IP de classe B, en binaire, ressemble à ceci :

10	xxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
Réseau			Ordinateurs	

Adressage IP de 128.xxx.xxx.xxx à 191.xxx.xxx.xxx

Masque de sous-réseau : 255.255.0.0

Elle contient 16 383 réseaux et 65 534 hôtes.

4-4-3) Classe C

Dans une adresse IP de classe C, les trois premiers octets représentent le réseau.

Les trois premiers bits sont 1,1 et 0, ce qui signifie qu'il y a 2^{21} possibilités de réseaux, c'est-à-dire 2097152.

Les réseaux disponibles en classe C sont donc les réseaux allant de **192.0.0.0** à **223.255.255.0**

L'octet de droite représente les ordinateurs du réseau, le réseau peut donc contenir : 2^8-2^1 = 254 ordinateurs.

Une adresse IP de classe C, en binaire, ressemble à ceci :

110	xxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
Réseau				Ordinateurs

Adressage IP de 192.xxx.xxx.xxx à 223.xxx.xxx.xxx

Masque de sous-réseau : 255.255.255.0

Elle contient 2 097 151 réseaux et 254 hôtes.

4-4-4) Classe D

Adressage IP de 224.xxx.xxx.xxx à 239.xxx.xxx.xxx

Elle est réservée à la multidiffusion (multicasting).

4-4-5) Classe E

Adressage IP de 240.xxx.xxx.xxx à 247.xxx.xxx.xxx

Elle est réservée à un usage futur.

4-5) Attribution des adresses IP

Le but de la division des adresses IP en trois classes A,B et C est de faciliter la recherche d'un ordinateur sur le réseau.

En effet avec cette notation il est possible de rechercher dans un premier temps le réseau que l'on désire atteindre puis de chercher un ordinateur sur celui-ci.

Ainsi l'attribution des adresses IP se fait selon la taille du réseau.

Classe	Nombre de réseaux possibles	Nombre d'ordinateurs maxi sur chacun
A	126	16777214
B	16384	65534
C	2097152	254

Les adresses de classe A sont réservées aux très grands réseaux, tandis que l'on attribuera les adresses de classe C à des petits réseaux d'entreprise par exemple

4-6) Adresses IP réservées

Il arrive fréquemment dans une entreprise ou une organisation qu'un seul ordinateur soit relié à internet, c'est par son intermédiaire que les autres ordinateurs du réseau accèdent à internet (on parle généralement de proxy ou de passerelle).

Dans ce cas de figure, seul l'ordinateur relié à internet a besoin de réserver une adresse IP auprès de l'ICANN. Toutefois, les autres ordinateurs ont tout de même besoin d'une adresse IP pour pouvoir communiquer ensemble en interne.

Ainsi, l'ICANN a réservé une poignée d'adresses dans chaque classe pour permettre d'affecter une adresse IP aux ordinateurs d'un réseau local relié à internet sans risquer de créer des conflits d'adresses IP sur le réseau des réseaux. Il s'agit des adresses suivantes:

➔ Adresses IP privées de classe A : 10.0.0.1 à 10.255.255.254, permettant la création de vastes réseaux privés comprenant des milliers d'ordinateurs.

➔ Adresses IP privées de classe B : 172.16.0.1 à 172.31.255.254, permettant de créer des réseaux privés de taille moyenne.

➔ Adresses IP privées de classe C : 192.168.0.1 à 192.168.0.254, pour la mise en place de petits réseaux privés.

4-7) Masques de sous-réseau

4-7-1) Notion de masque

Pour comprendre ce qu'est un masque, il peut-être intéressant de consulter la section «assembleur» qui parle du masquage en binaire

En résumé, on fabrique un masque contenant des 1 aux emplacements des bits que l'on désire conserver, et des 0 pour ceux que l'on veut annuler. Une fois ce masque créé, il suffit de faire un ET logique entre la valeur que l'on désire masquer et le masque afin de garder intacte la partie que l'on désire et annuler le reste.

Ainsi, un **masque réseau** (en anglais *netmask*) se présente sous la forme de 4 octets séparés par des points (comme une adresse IP), il comprend (dans sa notation binaire) des zéros aux niveau des bits de l'adresse IP que l'on veut annuler (et des 1 au niveau de ceux que l'on désire conserver).

4-7-2) Intérêt d'un masque de sous-réseau

Le premier intérêt d'un masque de sous-réseau est de permettre d'identifier simplement le réseau associé à une adresse IP.

En effet, le réseau est déterminé par un certain nombre d'octets de l'adresse IP (1 octet pour les adresses de classe A, 2 pour les adresses de classe B, et 3 octets pour la classe C). Or, un réseau est noté en prenant le nombre d'octets qui le caractérise, puis en complétant avec des 0. Le réseau associé à l'adresse *34.56.123.12* est par exemple *34.0.0.0*, car il s'agit d'une adresse IP de classe A.

Pour connaître l'adresse du réseau associé à l'adresse IP *34.56.123.12*, il suffit donc d'appliquer un masque dont le premier octet ne comporte que des 1 (soit 255 en notation décimale), puis des 0 sur les octets suivants.

Le masque est : *11111111.00000000.00000000.00000000*

Le masque associé à l'adresse IP *34.208.123.12* est donc *255.0.0.0*.

La valeur binaire de *34.208.123.12* est : *00100010.11010000.01111011.00001100*

Un ET logique entre l'adresse IP et le masque donne ainsi le résultat suivant :

00100010.11010000.01111011.00001100
ET
11111111.00000000.00000000.00000000
=
00100010.00000000.00000000.00000000

Soit *34.0.0.0*. Il s'agit bien du réseau associé à l'adresse *34.208.123.12*

En généralisant, il est possible d'obtenir les masques correspondant à chaque classe d'adresse:

➔ Pour une adresse de **Classe A**, seul le premier octet doit être conservé. Le masque possède la forme suivante *11111111.00000000.00000000.00000000*, c'est-à-dire **255.0.0.0** en notation décimale;

➔ Pour une adresse de **Classe B**, les deux premiers octets doivent être conservé, ce qui donne le masque suivant *11111111.11111111.00000000.00000000*, correspondant à **255.255.0.0** en notation décimale;

➔ Pour une adresse de **Classe C**, avec le même raisonnement, le masque possédera la forme suivante *11111111.11111111.11111111.00000000*, c'est-à-dire **255.255.255.0** en notation décimale

4-7-3) Création de sous-réseaux

Reprenons l'exemple du réseau 34.0.0.0, et supposons que l'on désire que les deux premiers bits du deuxième octet permettent de désigner le réseau.

Le masque à appliquer sera alors :

11111111.11000000.00000000.00000000

C'est-à-dire 255.192.0.0

Si on applique ce masque, à l'adresse 34.208.123.12 on obtient :

34.192.0.0

En réalité il y a 4 cas de figures possibles pour le résultat du masquage d'une adresse IP d'un ordinateur du réseau 34.0.0.0

➔ Soit les deux premiers bits du deuxième octet sont **00**, auquel cas le résultat du masquage est **34.0.0.0**

➔ Soit les deux premiers bits du deuxième octet sont **01**, auquel cas le résultat du masquage est **34.64.0.0**

➔ Soit les deux premiers bits du deuxième octet sont **10**, auquel cas le résultat du masquage est **34.128.0.0**

➔ Soit les deux premiers bits du deuxième octet sont **11**, auquel cas le résultat du masquage est **34.192.0.0**

Ce masquage divise donc un réseau de classe A (pouvant admettre 16 777 214 ordinateurs) en 4 sous-réseaux - d'où le nom de *masque de sous-réseau* - pouvant admettre 2^{22} ordinateurs, c'est-à-dire 4 194 304 ordinateurs.

Il peut être intéressant de remarquer que dans les deux cas, le nombre total d'ordinateurs est le même, soit 16 777 214 ordinateurs (4 x 4194304 - 2 = 16777214). Le nombre de sous-réseaux dépend du nombre de bits attribués en plus au réseau (ici 2).

Le nombre de sous-réseaux est donc :

Nombre de bits	Nombre de sous-réseaux
1	2
2	4
3	8
4	16
5	32
6	64
7	128
8 (impossible pour une classe C)	256

5) Le routage :

5-1) Les routeurs

Les routeurs sont les dispositifs permettant de "choisir" le chemin que les datagrammes vont emprunter pour arriver à destination.
Il s'agit de machines ayant plusieurs cartes réseau dont chacune est reliée à un réseau différent. Ainsi, dans la configuration la plus simple, le routeur n'a qu'à "regarder" sur quel réseau se trouve un ordinateur pour lui faire parvenir les datagrammes en provenance de l'expéditeur.

Toutefois, sur Internet le schéma est beaucoup plus compliqué pour les raisons suivantes :

➔ Le nombre de réseau auxquels un routeur est connecté est généralement important

➔ Les réseaux auxquels le routeur est relié peuvent être reliés à d'autres réseaux que le routeur ne connaît pas directement

Ainsi, les routeurs fonctionnent grâce à des tables de routage et des protocoles de routage, selon le modèle suivant :

➔ Le routeur reçoit une trame provenant d'une machine connectée à un des réseaux auquel il est rattaché

➔ Les datagrammes sont transmis à la couche IP

➔ Le routeur regarde l'en-tête du datagramme

➔ Si l'adresse IP de destination appartient à l'un des réseaux auxquels une des interfaces du routeur est rattaché, l'information doit être envoyée à la couche 4 après que l'en-tête IP ait été désencapsulée (enlevée)

➔ Si l'adresse IP de destination fait partie d'un réseau différent, le routeur consulte sa table de routage, une table qui définit le chemin à emprunter pour une adresse donnée

➔ Le routeur envoie le datagramme grâce à la carte réseau reliée au réseau sur lequel le routeur décide d'envoyer le paquet

Ainsi, il y a deux scénarios, soit l'émetteur et le destinataire appartiennent au même réseau auquel cas on parle de *remise directe*, soit il y a au moins un routeur entre l'expéditeur et le destinataire, auquel cas on parle de *remise indirecte*.

Dans le cas de la remise indirecte, le rôle du routeur, notamment celui de la table de routage, est très important. Ainsi le fonctionnement d'un routeur est déterminé par la façon selon laquelle cette table de routage est créée.

➔ Si la table routage est entrée manuellement par l'administrateur, on parle de **routage statique** (viable pour de petits réseaux)

➔ Si le routeur construit lui-même la table de routage en fonctions des informations qu'il reçoit (par l'intermédiaire de protocoles de routage), on parle de **routage dynamique**

5-2) La table de routage

La table de routage est une table de correspondance entre l'adresse de la machine visée et le noeud suivant auquel le routeur doit délivrer le message. En réalité il suffit que le message soit délivré sur le réseau qui contient la machine, il n'est donc pas nécessaire de stocker l'adresse IP complète de la machine: seul l'identificateur du réseau de l'adresse IP (c'est-à-dire l'ID réseau) a besoin d'être stocké.

La table de routage est donc un tableau contenant des paires d'adresses :

Adresse de destination	Adresse du prochain routeur directement accessible	Interface

Ainsi grâce à cette table, le routeur, connaissant l'adresse du destinataire encapsulée dans le message, va être capable de savoir sur quelle interface envoyer le message (cela revient à savoir quelle carte réseau utiliser), et à quel routeur, directement accessible sur le réseau auquel cette carte est connectée, remettre le datagramme.
Ce mécanisme consistant à ne connaître que l'adresse du prochain maillon menant à la destination est appelé *routage par sauts successifs* (en anglais *next-hop routing*).

Cependant, il se peut que le destinataire appartienne à un réseau non référencé dans la table de routage. Dans ce cas, le routeur utilise un **routeur par défaut** (appelé aussi *passerelle par défaut*).

Voici, de façon simplifiée, ce à quoi pourrait ressembler une table de routage :

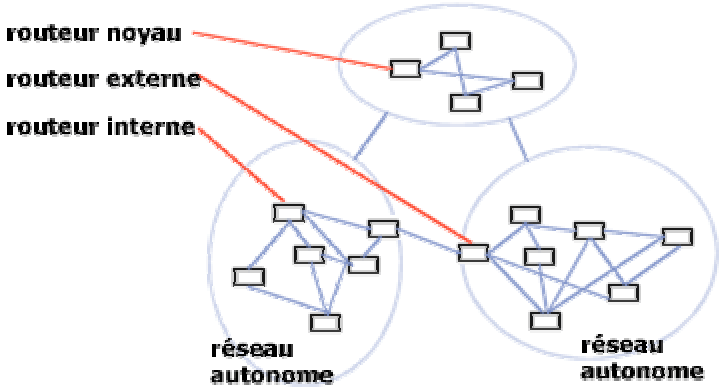
Adresse de destination	Adresse du prochain routeur directement accessible	Interface
194.56.32.124	131.124.51.108	2
110.78.202.15	131.124.51.108	2
53.114.24.239	194.8.212.6	3
187.218.176.54	129.15.64.87	1

Le message est ainsi remis de routeur en routeur par sauts successifs, jusqu'à ce que le destinataire appartienne à un réseau directement connecté à un routeur. Celui-ci remet alors directement le message à la machine visée...

Dans le cas du routage statique, c'est l'administrateur qui met à jour la table de routage.
Dans le cas du routage dynamique, par contre, un protocole appelé **protocole de routage** permet la mise à jour automatique de la table afin qu'elle contienne à tout moment la route optimale.

5-3) Les protocoles de routage

Internet est un ensemble de réseaux connectés. Par conséquent tous les routeurs ne font pas le même travail selon le type de réseau sur lequel ils se trouvent.



En effet, il y a différents niveaux de routeurs, ceux-ci fonctionnent donc avec des protocoles différents :

➔ Les **routeurs noyaux** sont les routeurs principaux car ce sont eux qui relient les différents réseaux

➔ Les **routeurs externes** permettent une liaison des réseaux autonomes entre eux. Ils fonctionnent avec un protocole appelé EGP (Exterior Gateway Protocol) qui évolue petit à petit en gardant la même appellation

➔ Les **routeurs internes** permettent le routage des informations à l'intérieur d'un réseau autonome. Ils s'échangent des informations grâce à des protocoles appelés IGP (Interior Gateway Protocol), tels que RIP et OSPF

5-3-1) Le protocole RIP

RIP signifie *Routing Information Protocol* (protocole d'information de routage). Il s'agit d'un protocole de type *Vector Distance* (Vecteur Distance), c'est-à-dire que chaque routeur communique aux autres routeurs la distance qui les sépare (le nombre de saut qui les sépare). Ainsi, lorsqu'un routeur reçoit un de ces messages il incrémente cette distance de 1 et communique le message aux routeurs directement accessibles. Les routeurs peuvent donc conserver de cette façon la route optimale d'un message en stockant l'adresse du routeur suivant dans la table de routage de telle façon que le nombre de saut pour atteindre un réseau soit minimal. Toutefois ce protocole ne prend en compte que la distance entre deux machines en termes de saut, mais il ne considère pas l'état de la liaison afin de choisir la meilleure bande passante possible.

5-3-2) Le protocole OSPF

OSPF (*Open Shortest Path First*) est plus performant que RIP et commence donc à le remplacer petit à petit. Il s'agit d'un protocole de type *protocole route-link* (que l'on pourrait traduire par *Protocole d'état des liens*), cela signifie que, contrairement à RIP, ce protocole n'envoie pas aux routeurs adjacents le nombre de sauts qui les sépare, mais l'état de la liaison qui les sépare. De cette façon, chaque routeur est capable de dresser une carte de l'état du réseau et peut par conséquent choisir à tout moment la route la plus appropriée pour un message donné.

De plus, ce protocole évite aux routeurs intermédiaires d'avoir à incrémenter le nombre de sauts, ce qui se traduit par une information beaucoup moins abondante, ce qui permet d'avoir une meilleure bande passante utile qu'avec RIP.

6) Quelques notions de ports :

6-1) L'utilité des ports

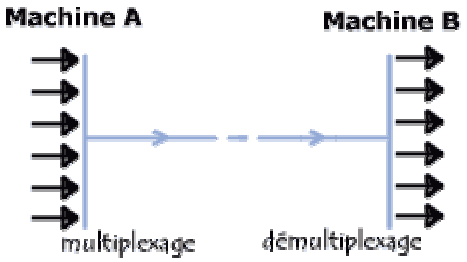
De nombreux programmes TCP/IP peuvent être exécutés simultanément sur Internet (vous pouvez par exemple ouvrir plusieurs navigateurs simultanément ou bien naviguer sur des pages HTML tout en téléchargeant un fichier par FTP). Chacun de ces programmes travaille avec un protocole, toutefois l'ordinateur doit pouvoir distinguer les différentes sources de données.

Ainsi, pour faciliter ce processus, chacune de ces applications se voit attribuer une adresse unique sur la machine, codée sur 16 bits : **un port** (la combinaison *adresse IP* + *port* est alors une adresse unique au monde, elle est appelée socket).

L'adresse IP sert donc à identifier de façon unique un ordinateur sur le réseau tandis que le numéro de port indique l'application à laquelle les données sont destinées. De cette manière, lorsque l'ordinateur reçoit des informations destinées à un port, les données sont envoyées vers l'application correspondante. S'il s'agit d'une requête à destination de l'application, l'application est appelée application **serveur**. S'il s'agit d'une réponse, on parle alors d'application **cliente**.

6-2) La fonction de multiplexage

Le processus qui consiste à pouvoir faire transiter sur une connexion des informations provenant de diverses applications s'appelle le multiplexage. De la même façon le fait d'arriver à mettre en parallèle (donc répartir sur les diverses applications) le flux de données s'appelle le **démultiplexage**.



Ces opérations sont réalisées grâce au port, c'est-à-dire un numéro associé à un type d'application, qui, combiné à une adresse IP, permet de déterminer de façon unique une application qui tourne sur une machine donnée.

6-3) Assignations par défaut

Il existe des milliers de ports (ceux-ci sont codés sur 16 bits, il y a donc 65536 possibilités), c'est pourquoi une assignation standard a été mise au point par l'**IANA** (*Internet Assigned Numbers Authority*), afin d'aider à la configuration des réseaux.

➔ Les ports 0 à 1023 sont les «**ports reconnus**» ou réservés («**Well Known Ports**»). Ils sont, de manière générale, réservés aux processus système (démons) ou aux programmes exécutés par des utilisateurs privilégiés. Un administrateur réseau peut néanmoins lier des services aux ports de son choix.

➔ Les ports 1024 à 49151 sont appelés «**ports enregistrés**» («**Registered Ports**»).

➔ Les ports 49152 à 65535 sont les «**ports dynamiques et/ou privés**» («**Dynamic and/or Private Ports**»).

Voici certains des ports reconnus les plus couramment utilisés :

Port	Service ou Application
21	FTP
23	Telnet
25	SMTP
53	Domain Name System
63	Whois
70	Gopher
79	Finger
80	HTTP
110	POP3
119	NNTP

Ainsi, un serveur (un ordinateur que l'on contacte et qui propose des services tels que FTP, Telnet, ...) possède des numéros de port fixes auxquels l'administrateur réseau a associé des services. Ainsi, les ports d'un serveur sont généralement compris entre 0 et 1023 (fourchette de valeurs associées à des services connus).

Du côté du client, le port est choisi aléatoirement parmi ceux disponibles par le système d'exploitation. Ainsi, les ports du client ne seront jamais compris entre 0 et 1023 caractères cet intervalle de valeurs représente les *ports connus*.

6-4) Plus d'informations

➔ Numéros de port assignés par l'IANA (Internet Assigned Numbers Authority)

7) Notions de VLAN :

7-1) Introduction aux VLAN

Un **VLAN** (*Virtual Local Area Network* ou *Virtual LAN*, en français *Réseau Local Virtuel*) est un réseau local regroupant un ensemble de machines de façon logique et non physique.

En effet dans un réseau local la communication entre les différentes machines est régie par l'architecture physique. Grâce aux réseaux virtuels (VLANs) il est possible de s'affranchir des limitations de l'architecture physique (contraintes géographiques, contraintes d'adressage, ...) en définissant une segmentation logique (logicielle) basée sur un regroupement de machines grâce à des critères (adresses MAC, numéros de port, protocole, etc.).

7-2) Typologie de VLAN

Plusieurs types de VLAN sont définis, selon le critère de commutation et le niveau auquel il s'effectue :

➔ Un **VLAN de niveau 1** (aussi appelés **VLAN par port**, en anglais *Port-Based VLAN*) définit un réseau virtuel en fonction des ports de raccordement sur le commutateur;

➔ Un **VLAN de niveau 2** (également appelé **VLAN MAC**, *VLAN par adresse IEEE* ou en anglais *MAC Address-Based VLAN*) consiste à définir un réseau virtuel en fonction des adresses MAC des stations. Ce type de VLAN est beaucoup plus souple que le VLAN par port car le réseau est indépendant de la localisation de la station;

➔ Un **VLAN de niveau 3** : on distingue plusieurs types de VLAN de niveau 3 :
- Le **VLAN par sous-réseau** (en anglais *Network Address-Based VLAN*) associe des sous-réseaux selon l'adresse IP source des datagrammes. Ce type de solution apporte une grande souplesse dans la mesure où la configuration des commutateurs se modifient automatiquement en cas de déplacement d'une station. En contrepartie une légère dégradation de performances peut se faire sentir dans la mesure où les informations contenues dans les paquets doivent être analysées plus finement.

- Le **VLAN par protocole** (en anglais *Protocol-Based VLAN*) permet de créer un réseau virtuel par type de protocole (par exemple TCP/IP, IPX, AppleTalk, etc.), regroupant ainsi toutes les machines utilisant le même protocole au sein d'un même réseau.

7-3) Les avantages du VLAN

Le VLAN permet de définir un nouveau réseau au-dessus du réseau physique et à ce titre offre les avantages suivants :

➔ Plus de souplesse pour l'administration et les modifications du réseau car toute l'architecture peut être modifiée par simple paramètrage des commutateurs

➔ Gain en sécurité car les informations sont encapsulées dans un niveau supplémentaire et éventuellement analysées

➔ Réduction de la diffusion du traffic sur le réseau

7-4) Plus d'informations

Les VLAN sont définis par les standards IEEE 802.1D, 802.1p, 802.1Q et 802.10. Pour plus d'information il est donc conseillé de se reporter aux documents suivants :

➔ IEEE 802.1D

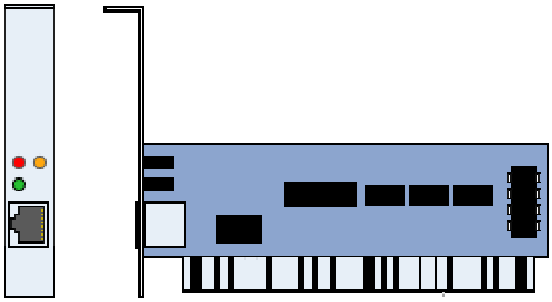
➔ IEEE 802.1Q

➔ IEEE 802.10

8) L'adresse matérielle (adresse MAC) :

8-1) Qu'est-ce qu'une carte réseau ?

La **carte réseau** (appelée *Network Interface Card* en anglais et notée **NIC**) constitue l'interface entre l'ordinateur et le câble du réseau. La fonction d'une carte réseau est de préparer, d'envoyer et de contrôler les données sur le réseau.



La carte réseau possède généralement deux témoins lumineux (LEDs) :

➔ La LED verte correspond à l'alimentation de la carte ;

➔ La LED orange (10 Mb/s) ou rouge (100 Mb/s) indique une activité du réseau (envoi ou réception de données).

Pour préparer les données à envoyer, la carte réseau utilise un **transceiver** qui transforme les données parallèles en données séries. Chaque carte dispose d'une adresse unique, appelée **adresse MAC**, affectée par le constructeur de la carte, ce qui lui permet d'être identifiée de façon unique dans le monde parmi toutes les autres cartes réseau.
Les cartes réseau disposent de paramètres qu'il est possible de configurer. Parmi eux figurent l'interruption matérielle (IRQ), l'adresse de base du port E/S et l'adresse de base de la mémoire (DMA).

Pour garantir la compatibilité entre l'ordinateur et le réseau, la carte doit être adaptée à l'architecture du bus de données de l'ordinateur et avoir le type de connecteur approprié au câblage. Chaque carte est conçue pour s'adapter à un certain type de câble. Certaines cartes comprennent plusieurs connecteurs d'interfaces (à paramétrer soit avec les cavaliers, soit avec les DIP, soit de façon logicielle). Les connecteurs les plus répandus sont les connecteurs RJ-45.

NB : Certaines topologies réseau propriétaires utilisant la paire torsadée ont recours au connecteur RJ-11. Ces topologies sont parfois appelées « *pré-10BaseT* ».

Enfin pour garantir cette compatibilité entre ordinateur et réseau, la carte doit être compatible avec la structure interne de l'ordinateur (architecture du bus de données) et avoir un connecteur adapté à la nature du câblage.

8-2) Quel est le rôle de la carte réseau ?

Une carte réseau sert d'interface physique entre l'ordinateur et le câble. Elle prépare pour le câble réseau les données émises par l'ordinateur, les transfère vers un autre ordinateur et contrôle le flux de données entre l'ordinateur et le câble. Elle traduit aussi les données venant du câble et les traduit en octets afin que l'Unité Centrale de l'ordinateur les comprenne. Ainsi une carte réseau est une carte d'extension s'insérant dans un connecteur d'extensions (slot).

8-3) La préparation des données

Les données se déplacent dans l'ordinateur en empruntant des chemins appelés « bus ». Plusieurs chemins côte à côte font que les données se déplacent en parallèle et non en série (les unes à la suite des autres).

➔ Les premiers bus fonctionnaient en 8 bits (8 bits de données transportés à la fois)

➔ L'ordinateur PC/AT d'IBM introduit les premiers bus 16 bits

➔ Aujourd'hui, la plupart des bus fonctionnent en 32 bits

Toutefois sur un câble les données circulent en série (un seul flux de bits), en se déplaçant dans un seul sens. L'ordinateur peut envoyer **OU** recevoir des informations mais il ne peut pas effectuer les deux simultanément. Ainsi, la carte réseau restructure un groupe de données arrivant en parallèle en données circulant en série (1 bit).
Pour cela, les signaux numériques sont transformés en signaux électriques ou optiques susceptibles de voyager sur les câbles du réseau. Le dispositif chargé de cette traduction est le **Transceiver**.

8-4) Le rôle d'identificateur

➔ La carte traduit les données et indique son adresse au reste du réseau afin de pouvoir être distinguée des autres cartes du réseau.

➔ Adresses MAC : définies par l'IEEE (Institute of Electrical and Electronics Engineer) qui attribue des plages d'adresses à chaque fabriquant de cartes réseau.

➔ Elles sont inscrites sur les puces des cartes : procédure appelée «Gravure de l'adresse sur la carte». Par conséquent, chaque carte a une adresse MAC UNIQUE sur le réseau.

8-5) Les autres fonctions de la carte réseau

L'ordinateur et la carte doivent communiquer afin que les données puissent passer de l'un vers l'autre. L'ordinateur affecte ainsi une partie de sa mémoire aux cartes munies d'un Accès Direct à la Mémoire (DMA : Direct Access Memory).
La carte indique qu'un autre ordinateur demande des données à l'ordinateur qui la contient. Le bus de l'ordinateur transfère les données depuis la mémoire de l'ordinateur vers la carte réseau.

Si les données circulent plus vite que la carte ne peut les traiter, elles sont placées dans la mémoire tampon affectée à la carte (RAM) dans laquelle elles sont stockées temporairement pendant l'émission et la réception des données.

8-6) Envoi et contrôle des données

Avant que la carte émettrice envoie les données, elle dialogue électroniquement avec la carte réceptrice pour s'accorder sur les points suivants :

➔ Taille maximale des groupes de données à envoyer

➔ Volume de données à envoyer avant confirmation

➔ Intervalles de temps entre les transmissions partielles de données

➔ Délai d'attente avant envoi de la confirmation

➔ Quantité que chaque carte peut contenir avant débordement

➔ Vitesse de transmission des données

Si une carte plus récente, donc plus perfectionnée, communique avec une carte plus lente, elles doivent trouver une vitesse de transmission commune. Certaines cartes ont des circuits leur permettant de s'adapter au débit d'une carte plus lente.
Il y a donc acceptation et ajustement des paramètres propres à chacune des deux cartes avant émission et réception des données.

8-7) Paramètres de configuration de la carte

Les cartes réseau sont munies d'options de configuration. Entre autres :

➔ Interruption (IRQ) : Dans la plupart des cas, ce sont les IRQ 3 et 5 qui sont attribués aux cartes réseau. L'IRQ 5 est même conseillé (s'il est disponible !) et la plupart des cartes l'utilisent comme paramètre par défaut.

➔ Adresse de base du port d'entrée/sortie (E/S) : Chaque périphérique doit utiliser une adresse de base différente pour le port correspondant.

➔ Adresse de base de la mémoire : Elle désigne un emplacement de la mémoire vive (RAM) de l'ordinateur. La carte utilise cet emplacement comme tampon pour les données qui entrent et qui sortent. Ce paramètre est parfois appelé « adresse de début » (RAM Start Address). En général, l'adresse de base de la mémoire pour une carte réseau est D8000. Le dernier 0 est parfois supprimé pour certaine carte réseau. Il est essentiel de prendre soin de ne pas sélectionner une adresse de base déjà utilisée par un autre périphérique. A noter toutefois que certaines cartes réseau n'ont pas de réglage pour l'adresse de base de la mémoire car elles n'utilisent pas les adresses RAM de la machine.

➔ **Le transceiver**

Remarque : il est possible de configurer la carte de manière logicielle. Les paramètres doivent correspondre avec la disposition des cavaliers ou des commutateurs DIP (Dual Inline Package) situés sur la carte réseau. Les réglages sont fournis avec la documentation de la carte. Beaucoup de cartes récentes sont en PnP (Plug and Play). Cela dispense de configurer la carte à la main mais peut parfois être gênant (apparition de conflits) auquel cas il est généralement agréable de pouvoir désactiver l'option PnP et configurer la carte "à la main".