مكتب التكوين المهني وإنعاش الشغل
Office de la Formation Professionnelle et de la Promotion du Travail

Le protocole TCP-IP

OFPPT
DIRECTION RECHERCHE ET INGENIERIE DE FORMATION
SECTEUR NTIC

Sommaire

TCP / IP

I - TCP/IP et les réseaux

I-A- Pourquoi un protocole ?

Un **protocole de communication** est un ensemble de règles permettant à plusieurs ordinateurs, éventuellement sur des réseaux physiques différents et utilisant des OS différents, de dialoguer entre eux.
Ainsi grâce à **TCP/IP**, des ordinateurs sous UNIX et sur un réseau Ethernet peuvent dialoguer avec des ordinateurs sous NT sur un réseau Token-Ring.
TCP/IP peut fonctionner :

- sur des réseaux locaux physiques de type Ethernet, Fast Ethernet, Token-Ring, FDDI
- sur des réseaux de type WAN comme ATM, LAPB ou des liaisons par RTC ou LS.

I-B- Rappel du modèle OSI de l'ISO

I- B- 1- Les couches

Le modèle **OSI**[1] de l'**ISO**[2] permet de définir un modèle pour des ordinateurs communicants. Tout ordinateur conforme à ce modèle peut dialoguer avec ces homologues en utilisant le même "langage" et les mêmes méthodes de communication.

- Le modèle est composé de 7 couches.
- Chaque couche assure une fonction bien déterminée.
- Chaque couche utilise les services de la couche inférieure. Par exemple la couche **Réseau** utilise les services de la couche **Liaison** qui utilise elle-même les services de la couche **Physique**.
- Chaque couche possède un point d'entrée pour les services offerts, nommé SAP =Service Access Point. Ainsi la couche **Session** possède un point d'accès **SSAP** et la couche **Transport** un point d'accès **TSAP**.
- Chaque couche d'un ordinateur dialogue avec la couche homologue d'un autre ordinateur en utilisant un **protocole** spécifique à la couche (Données de protocole = PDU =Protocol Data Unit).

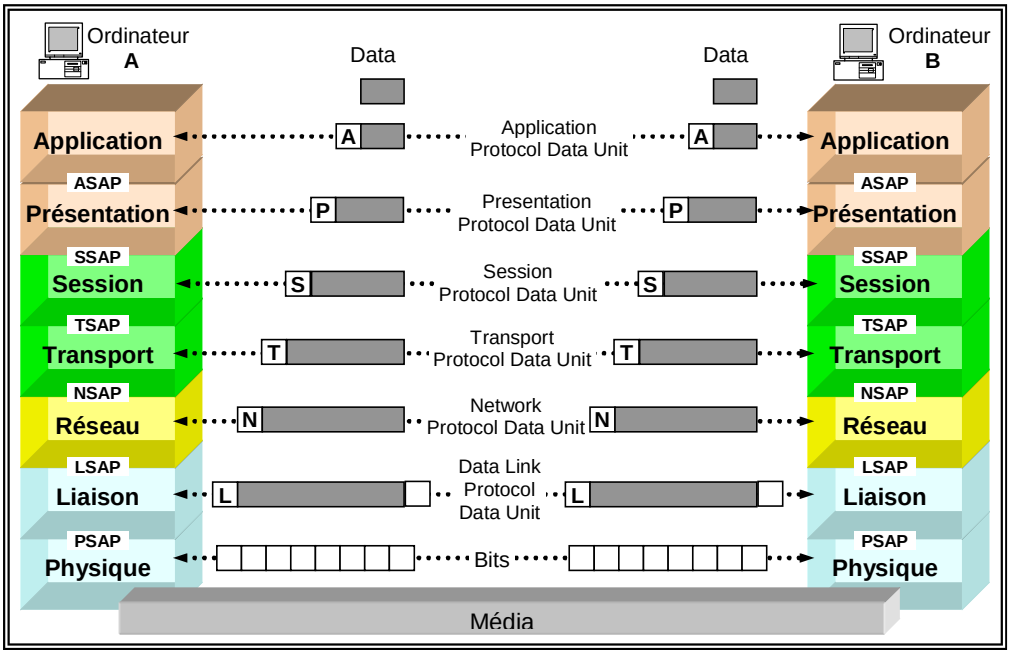
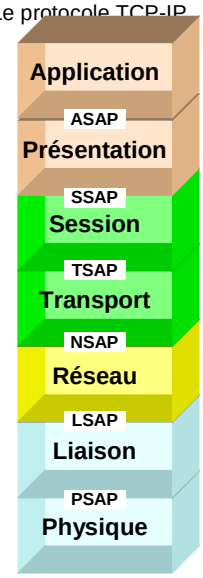


Figure I-1 : Modèle OSI de l'ISO.

I- B- 2- Rôle des couches

[1] OSI = Open System Interconnection = Interconnexion des systèmes ouverts.
[2] ISO = International Organization of Standards.



La couche **Application** fournit les services de communication aux applications utilisateurs. Par exemple les services de transferts de fichiers, gestion de message pour les applications de messagerie, accès à des bases de données
La couche **Présentation** gère la représentation des données. La couche Présentation utilise un langage commun compréhensible par tous les nœuds du réseau.
La couche **Session** gère les connexions entre les applications homologues sur les machines en réseau. Elle assure l'initialisation et la clôture des sessions ainsi que les reprises en cas d'incident.
La couche **Transport** garantit que les données reçues sont strictement conformes à celles qui ont été émises. Cette couche assure le multiplexage sur une seule liaison physique et la remise des données à l'application par un numéro de port.
La couche **Réseau** assure le cheminement des données sous forme de paquets dans l'internet. Ceci est réalisé par un système d'adressage spécifique à cette couche et utilisé par les routeurs de l'internet.
La couche **Liaison** prend en charge les données de la couche Physique. Elle gère des trames (Ethernet, Token-Ring, PPP,…), les adresses du réseau Physique, la méthode d'accès au réseau physique et contrôle la validité des trames transmises.
La couche **Physique** transmet et reçoit des bits sur le média sous forme d'un signal électrique. Elle assure l'encodage et désencodage de ces bits.

Figure I-2 : Le rôle des couches du modèle OSI de l'ISO.

I- B- 3- Les sous couches de l'IEEE

Le monde des réseaux locaux possède un organisme de standardisation qui lui est propre. Il s'agit de l'**IEEE**[3]. Cet organisme gère les couches qui sont exclusives aux réseaux locaux. L'**IEEE** divise en deux la couche liaison de données du modèle OSI de l'ISO. Ces deux sous-couches sont :

- La couche **MAC** -- Media Access Control

Cette couche concerne les méthodes d'accès au support du réseau local. Ainsi Ethernet correspond à la norme IEEE 802.3, alors que Token-Ring est concernée par la norme IEEE 802.5

- La couche **LLC** -- Logical Link Control

Tous les types de réseaux définis au niveau de la sous-couche MAC possèdent une interface commune avec la couche **Réseau**, c'est-à-dire avec les protocoles utilisés sur le réseau. Ceci permet d'utiliser n'importe quel protocole avec n'importe quel type de réseau physique. Cette couche est responsable de la transmission des données entre les nœuds du réseau. Elle fournit des services de datagramme en mode connecté ou non connecté ou des services de circuits virtuels.

- Dans le mode **datagramme**, les paquets générés par la couche contiennent une adresse source et une adresse destination. Aucun chemin n'est établi par avance et les paquets peuvent passer par des chemins différents. Aucune vérification n'est assurée tant qu'au séquencement des paquets à leur arrivée.
- Dans le mode **circuit virtuel**, une connexion est établie entre les nœuds communicants ainsi qu'un contrôle du séquencement et de la validité des trames transmises. Un contrôle de flux est aussi assuré.

  La couche **LLC** peut assurer trois types de services aux couches supérieures:
  - **Type 1** : **Service de datagramme** sans accusé de réception en mode point à point, multipoint ou diffusion.
  - **Type 2** : **Services de circuits virtuels**. Assure les services de séquencement, de contrôle de flux et de correction d'erreur.
  - **Type 3** : **Service de datagramme avec accusé de réception**.

[3] IEEE Institute of Electrical and Electronic Engineers.

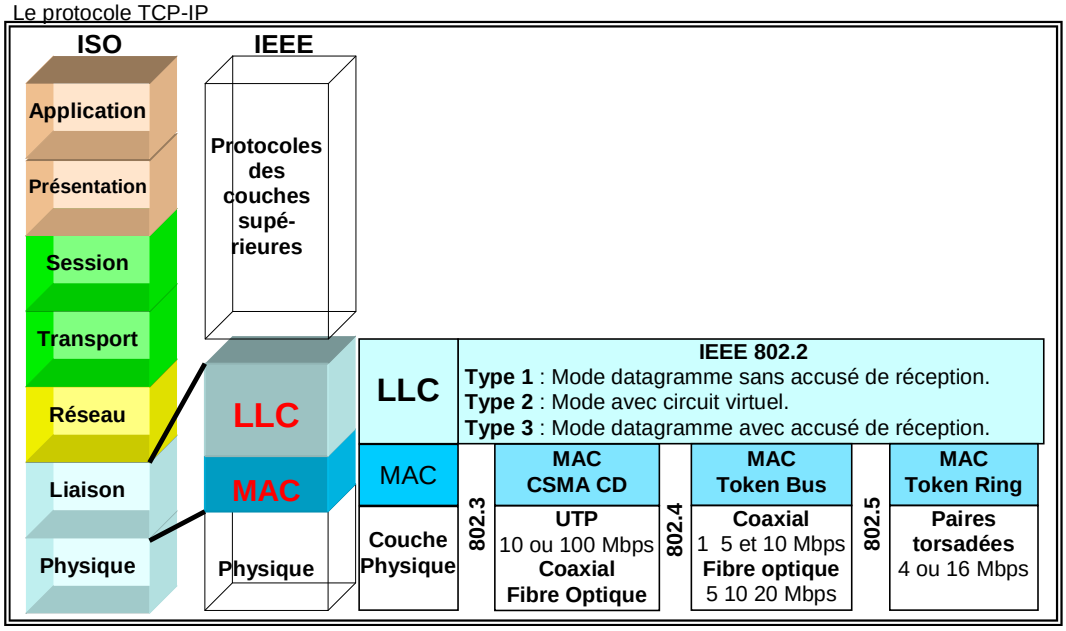


Figure I-3 : Sous-couches IEEE par rapport aux couches ISO

I-C- TCP/IP et le modèle DoD

TCP/IP est antérieur au modèle de l'ISO. Il est conforme au modèle **DoD**[4] . Ce modèle comporte 4 couches.

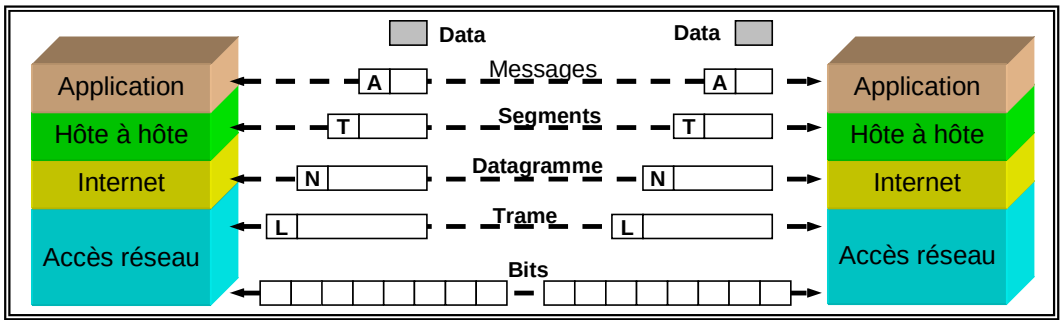


Figure I-4 : Modèle DoD.

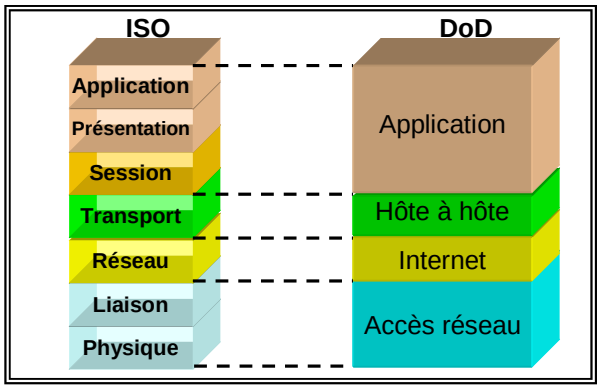


Figure I-5 : Comparaison modèle OSI et modèle DoD.

[4] DoD = Department of Defence.

II - Historique de TCP/IP

La nécessité de relier entre eux des réseaux de types différents, a conduit un organisme de la Défense américaine **DARPA**[1], à la fin des années 60, à créer un protocole ou plus exactement une **suite de protocoles** dénommée **TCP/IP**[6]. Les protocoles **TCP** et **IP** définissent un ensemble de formats et de règles pour l'émission et la réception de données indépendamment des types de réseau physique et d'ordinateurs utilisés. Les protocoles TCP/IP fortement implantés dans l'environnement UNIX, bien que non conformes au modèle de l'OSI, sont devenus des standards de fait.
Le réseau qui utilise TCP/IP est un réseau à commutation de paquets. Ce type de réseau transmet des informations sous forme de petits groupes d'octets appelés **Paquets**. Si un fichier doit être transmis, il est d'abord fragmenté en paquets à l'émission puis, le fichier est réassemblé en regroupant les paquets à la réception.

III - Inter-réseaux et Routage IP

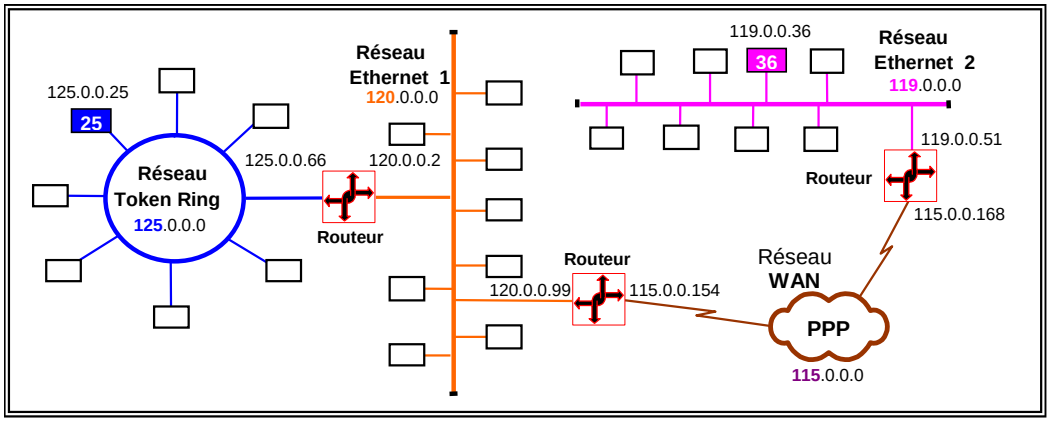


Figure III-1 : Un inter-réseau ou internet IP.

Dans la figure ci-dessus, imaginons que le nœud **25** du réseau **Token-Ring** désire envoyer des données au nœud **36** du réseau **Ethernet 2**. Le type de trame, la méthode d'accès, le système d'adressage et le débit du réseau Token-Ring sont incompatibles avec ceux du réseau Ethernet. Les données ne peuvent être transmises en l'état. Elles doivent, grâce aux **routeurs** du réseau, changer de type de trame à chaque nouveau type de réseau.
Un **système d'adressage**, *indépendant du type de réseau physique*, doit être utilisé pour désigner de façon unique chaque nœud sur l'inter-réseau. Le protocole **IP** possède ce type d'adressage composé d'une **adresse réseau** (NetID= Network ID) et d'une **adresse nœud** (HostID = Host ID) sur chaque réseau.
Par exemple, 125, 120, 115 et 119 désignent respectivement les adresses réseaux des réseaux Token-Ring, Ethernet 2, la liaison asynchrone en **PPP**[7] et Ethernet 1.
L'adresse **125**.0.0.25 désigne l'adresse du nœud 25 sur le réseau **125**. L'adresse **119**.0.0.36 désigne le nœud 36 sur le réseau **119** Ethernet 2.
Chaque **routeur** est équipé d'au moins 2 interfaces réseaux. Des tables de routage internes à chaque routeur permettent de connaître le chemin à emprunter pour transporter des données d'un nœud à un autre. Lorsque le paquet IP arrive dans une trame Token-Ring dans le routeur 1 à destination du nœud 36 du réseau Ethernet 2, celui-ci lit l'adresse IP de destination, et repositionne le paquet IP dans une trame Ethernet. Lorsque la trame parvient au routeur 2, le paquet est positionné dans des trames PPP. Puis lorsqu'il arrive au routeur 3, il est repositionné dans une trame Ethernet. L'adresse IP de destination n'a pas changé pendant tout le parcours, par contre, les adresses physiques (MAC) ont été modifiées sur chaque réseau.

[5] DARPA = Defence Advanced Research Projects Agency
[6] TCP = Transmission Control Protocol IP= Internet Protocol
[7] PPP = Point to Point Protocol

IV - Couches

IV-A- TCP/IP et les modèles ISO et DoD

La suite des protocoles appelée aussi **pile de protocoles**[8] **IP** ne correspond pas au **modèle OSI** de l'**ISO**, celui-ci a été normalisé en 1979, il est donc postérieur à la création de TCP/IP.
La pile de protocoles IP correspond au **modèle DoD** (Department of Defence). Le dessin suivant montre l'équivalence entre ces couches et les différents protocoles de la pile. Les protocoles correspondant aux couches 6 et 7 ISO sont des applications de transmissions qui s'appuient sur TCP/IP. Les couches 1 et 2 dépendent du type de réseau utilisé.
Tous les standards (normes) de la ***communauté Internet*** sont publiés sous forme de **RFC**[9]. Chacune est identifiée par un numéro et décrit le fonctionnement d'un protocole de la pile TCP/IP ou d'un matériel comme par exemple les routeurs IP.

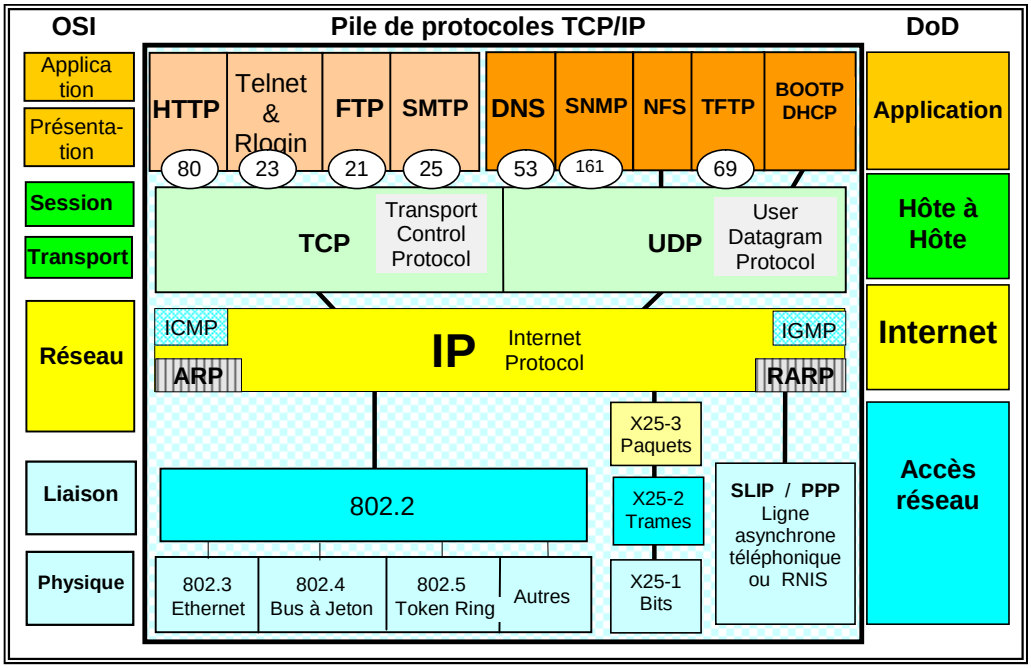


Figure IV-1 : TCP /IP et les couches OSI.

III- *Protocoles réseau*

Mise en forme : Puces et numéros

IP	Internet Protocol	Fournit les services de communication d'inter-réseau aux clients de la couche 4.
ARP	Address Resolution Protocol	Protocole permettant de faire correspondre une adresse IP à une adresse Physique.
RARP	Reverse ARP	Protocole inverse faisant correspondre une adresse Physique à une adresse IP.
ICMP	Internet Control Message Protocol	Contrôle la transmission des messages d'erreur et des messages entre hôtes, passerelles ou routeurs.
IGMP	Internet Group Management Protocol	Permet d'envoyer des datagrammes à un groupe de machines grâce à un adressage multicast.

Figure IV-2 : Protocoles réseau.

[8] Protocol Stack
[9] RFC = Request For Comments

Protocoles transport

TCP	Transmission Control Protocol	Protocole orienté connexion, fiable et à flot de données.
UDP	User Datagram Protocol	Protocole sans connexion, orienté transaction sans accusé de réception, parallèle à TCP.

Figure IV-3 : Protocoles de transport.

Applications au-dessus de TCP/IP

HTTP	HyperText Transfert Protocol	Service de distribution de pages en hypertexte sur des serveurs WEB.
Telnet	Telecommunications Network	Fournit un service d'émulation de Terminal.
Rlogin	Remote Login	Commande Unix permettant la connexion des terminaux à d'autres serveurs Unix du réseau.
FTP	File Transfert Protocol	Permet l'échange de fichiers complets entre ordinateurs.
SMTP	Simple Mail Transfer Protocol	Offre une fonction importante de messages de texte entre hôtes.
DNS	Domain Name Service	Propose des services de répertoires. DNS est un service complexe qui permet d'associer un nom et une adresse.
SNMP	Simple Network Management Protocol	Protocole de management de réseau.
NFS	Network File System	Système de fichiers en réseau.
TFTP	Trivial File Transfert Protocol	Protocole simplifié de transfert de fichiers utilisé principalement par les clients sans disque.
BOOTP **DHCP**	Bootstrap Protocol Dynamic Host Configuration Protocol	Protocoles fournissant une adresse IP de façon dynamique au démarrage des stations.

Figure IV-4 : Couches 6 et 7. Applications utilisées avec TCP / IP.

Vos notes personnelles :

V - Fonctionnement de la pile de protocoles IP

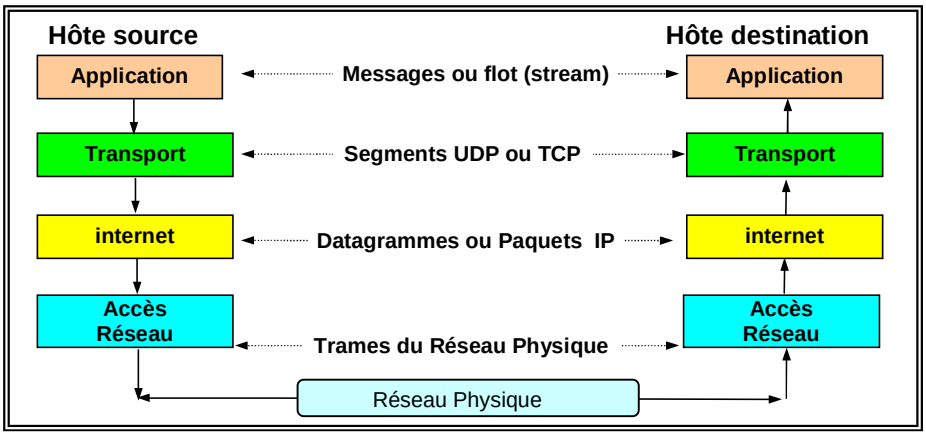


Figure V-1 : Fonctionnement de TCP/IP.

Les **applications** développées pour TCP/IP utilisent généralement plusieurs des protocoles de la suite. Elles communiquent avec la couche **transport**, elle-même communiquant avec les couches inférieures, pour aboutir au support physique qu'est le réseau. A destination, les couches inférieures repassent les informations aux couches supérieures pour aboutir à l'application de l'hôte destination.
Chaque couche de la pile remplit une fonction bien spécifique. Une couche quelconque rend des **Services** à la couche qui lui est immédiatement **supérieure**. Chaque couche de même niveau dans les ordinateurs Source et Destination dialogue avec son homologue. Ce dialogue est décrit dans le **protocole** correspondant à la couche. Par exemple **IP** pour la couche **internet** (ou **réseau**) et **TCP** pour la couche **transport**.

V-A- Encapsulation

Dans l'ordinateur qui émet des données, les couches communiquent avec les couches homologues de l'autre ordinateur. Chaque couche ajoute des informations nommées **en-têtes**, destinées à communiquer avec la couche homologue située dans l'ordinateur de l'autre extrémité. Chaque nouveau paquet ainsi formé est inséré dans un paquet de la couche inférieure. Cette opération s'appelle **encapsulation**.

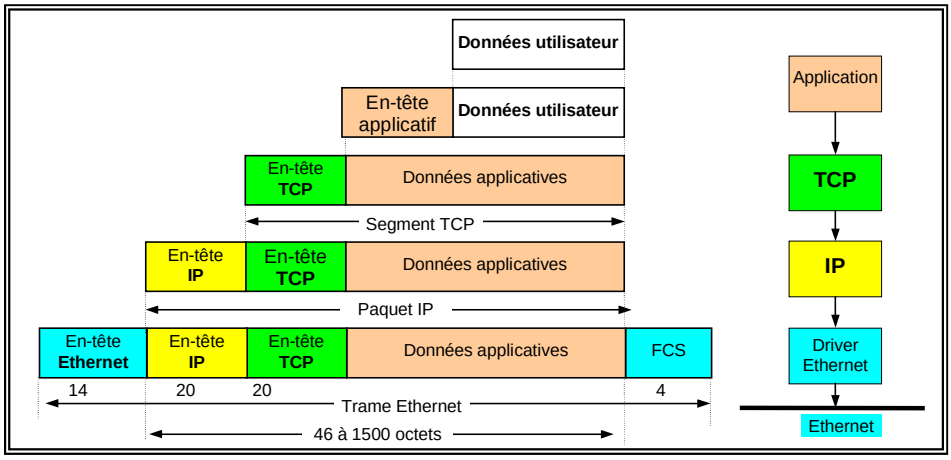


Figure V-2 : Encapsulation.

- Les données de l'**application**, avec leur en-tête sont passées à la couche **TCP** qui rajoute le sien. L'ensemble est appelé **segment TCP**.
- L'ensemble des données qu'envoie IP à la couche Ethernet est appelé **datagramme IP**.
- L'ensemble de bits structuré envoyé sur le réseau est une **trame Ethernet**.

L'ensemble des données inclus dans IP aurait pu être un **datagramme UDP**, si l'application utilisait ce type de protocole plutôt que TCP.

Le protocole TCP-IP
TCP et **UDP** utilisent des **numéros de ports** sur 16 bits pour connaître l'application qui leur a passé des données.
Les protocoles **ARP**, **RARP**, **ICMP** et **IGMP** attaquent directement le datagramme **IP**. Le champ ***type*** de cette trame permet de savoir quel est le protocole utilisé dans le champ de données.

V- A- 1- Encapsulation IP dans les diverses trames Ethernet

Le standard réseau Ethernet d'origine a été repris et modifié par le **Comité 802** de l'**IEEE**. Il existe donc plusieurs définitions de types de trames Ethernet. Quel que soit le type de trame, il faut cependant que les **paquets IP** puissent y être encapsulés. Dans la norme de l'IEEE la couche ***liaison*** est divisée en 2 sous-couches :

- la couche **MAC (802.3)** qui correspond à la gestion de CSMA/CD et à l'interfaçage avec la couche physique.
- la couche **LLC (802.2)** qui définit le contrôle de la liaison. Cette fonction est commune a tous les types de réseaux physiques.

La trame Ethernet_802.2 définie par l'IEEE contient des champs supplémentaires par rapport à la trame Ethernet. Ce sont les champs **DSAP** et **SSAP** (Destination et Source **Service Access Point**). Le champ contrôle contient la valeur 03 en hexadécimal.
Dans la trame Ethernet_SNAP, les 3 octets du champ **Org** sont à **0**. Le champ type contient la valeur qui est contenue dans le champ **Type** de la trame Ethernet. Cette valeur représente le code du protocole utilisé dans le champ de données de la trame Ethernet.

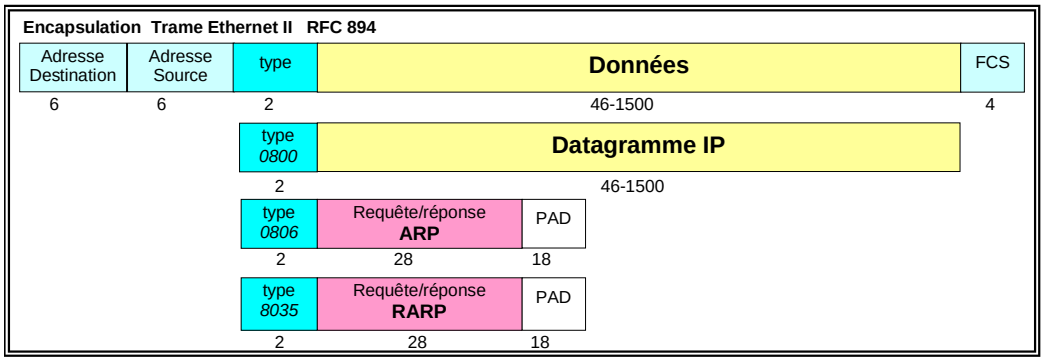


Figure V-3 : Encapsulation de IP dans des trames Ethernet II.

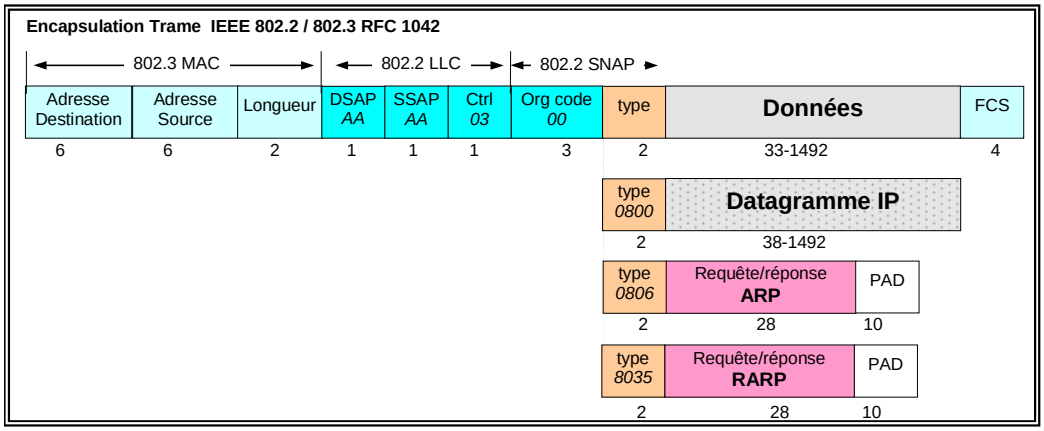


Figure V-4 : : Encapsulation de IP dans des trames Ethernet 802.2.

V-B- Multiplexage et Démultiplexage

V- B- 1- Multiplexage

Le champ "**Type**" dans une **trame Ethernet** permet d'indiquer le code des différents types de protocoles (IP, ARP et RARP). De même au niveau IP, le champ "**Type**" de **l'en-tête IP**, permet de transporter TCP ou UDP. Enfin au niveau transport, les **numéros de ports** indiquent les applications concernées. Cette propriété de mélanger les protocoles est appelée **multiplexage**

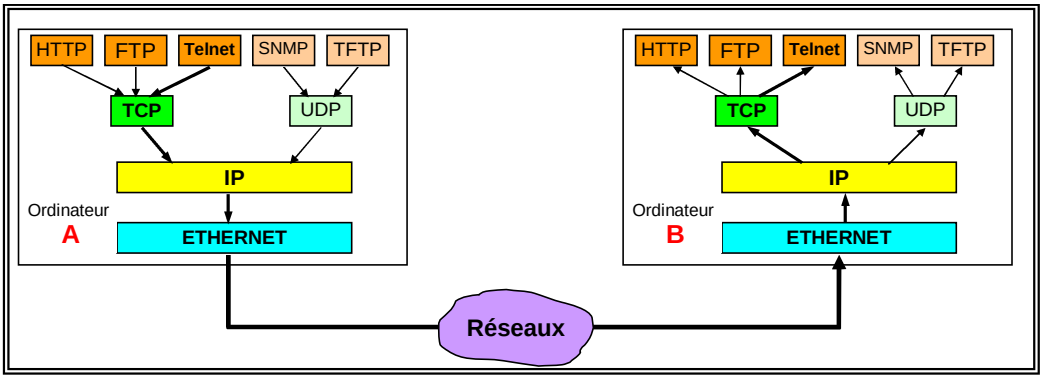


Figure V-5 : Multiplexage.

V- B- 2- Démultiplexage

A l'inverse lorsqu'une machine reçoit une trame Ethernet, les données applicatives doivent remonter jusqu'aux couches supérieures en traversant les couches basses. A chaque niveau, l'en-tête correspondant à la couche est interprété pour savoir à quel protocole ou applications les données doivent être remises. L'en-tête n'est pas transmis à la couche supérieure.

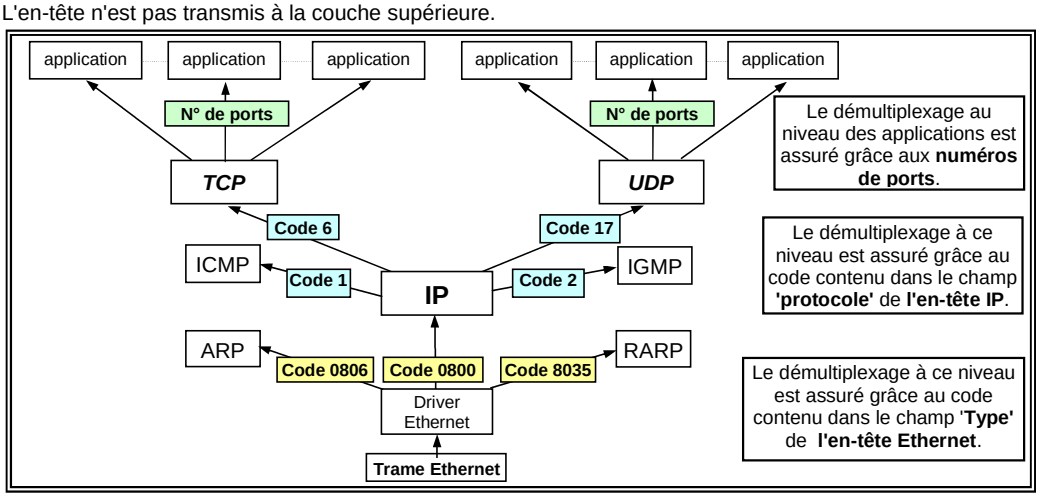


Figure V-6. Démultiplexage IP.

- La reconnaissance des datagrammes **IP**, **ARP** ou **RARP** est assurée par les codes **0800**, **0806** et **8035** contenus dans le champ '**type**' de la *trame Ethernet*.
- La reconnaissance des messages **ICMP** et **IGMP** est assurée par les codes **1** et **2** dans le champ type de l'***en-tête IP***. Les valeurs **6** et **17** indiquent qu'un segment **TCP** ou **UDP** suit l'***en-tête IP***.
- Les **numéros de ports** contenus dans les en-têtes **TCP** et **UDP** permettent de connaître l'**application** à laquelle il faut restituer les données.

Chaque application côté *serveur* utilise un numéro de port "**bien connu**" (well-know). Ainsi, l'application **Telnet** serveur utilise en principe le port **TCP 23** et **FTP** le port **TCP 21**, alors que **TFTP** serveur utilise le port **UDP 69**. Les numéros de port côté serveur sont compris entre **1** et **1023**.
Les applications côté *client* utilisent des "**ports éphémères**" dont les numéros sont compris entre **1024** et **5000**. La gestion de ces numéros de ports est complètement transparente pour les utilisateurs.
La liste des **Ports TCP et UDP** est contenue dans le fichier **Services** des ordinateurs travaillant sous IP.

VI - Adresses IP

Au niveau de la couche **Liaison**, les nœuds du réseau communiquent avec les autres stations en utilisant des adresses qui dépendent du type de réseau utilisé. Un nœud peut être un micro-ordinateur, un serveur de fichier, une imprimante réseau ou n'importe quel périphérique utilisant TCP/IP. Chaque nœud possède une adresse **physique** ou adresse **MAC**[10].
Dans les réseaux Ethernet et Token-Ring, l'adresse physique est contenue dans une ROM sur chaque interface réseau. Toutes les adresses sont différentes et comportent 6 octets. Cette adresse est déterminée par le constructeur de l'interface selon un plan de numérotation à l'échelle mondiale.
Dans le réseau X25, l'adresse déterminée par le concessionnaire du réseau comporte au maximum 15 chiffres décimaux.
Dans le réseau LocalTalk d'Apple, l'adresse comporte un octet pour déterminer le numéro du réseau et 2 pour déterminer le numéro de la station.

VI-A- Généralités

Les adresses **IP** au contraire sont des adresses logiques. Elles sont **indépendantes du type de réseau** utilisé. Dans la version 4 de IP, elles comportent toujours **32 bits**, dont une partie identifie le réseau (**NetID**), l'autre le nœud sur ce réseau (**HostID**).

VI- A- 1- Types d'adresses

- **Unicast** : Adresse permettant l'adressage d'une **seule** machine.
- **Multicast** : Adresse correspondant à un **groupe** de machines.
- **Broadcast** : Adresse correspondant à **toutes** les machines d'un réseau.

VI-B- Représentation des adresses IP

La représentation de cette adresse se fait dans une notation "**décimale pointée**" (dotted-decimal notation), c'est-à-dire que chaque octet de l'adresse est représenté par un nombre décimal, séparé du suivant par un point. Par exemple :

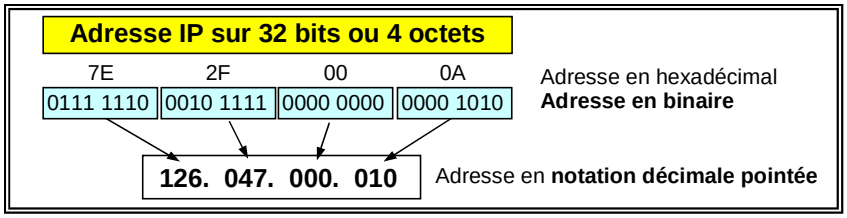


Figure VI-1 : Adresses IP. Notation décimale pointée.

Parfois, la représentation se fait en Hexadécimal de la façon suivante : **Øx7E.Øx2F.Øx00.Øx0A**

Vos notes :

_____
_____
_____
_____
_____
_____
_____
_____
_____

[10] MAC = Medium Access Control

VI-C- Classes d'adresses

Il existe 5 classes d'adresses IP.

VI- C- 1- Classe A

Dans cette classe, l'adresse réseau est définie sur **7 bits** et l'adresse hôte sur **24 bits**.

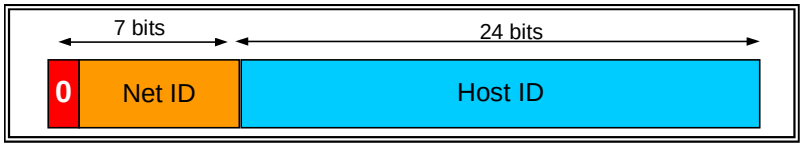


Figure VI-2 : Adressage IP Classe A.

VI- C- 2- Classe B

Dans cette classe, l'adresse réseau est sur 14 bits et l'adresse hôte sur 16 bits.

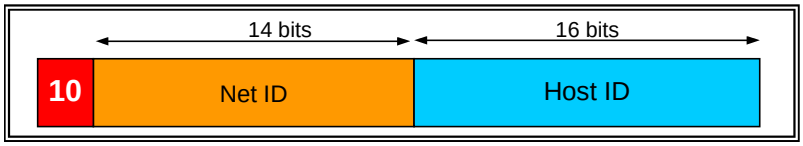


Figure VI-3 : Adressage IP Classe B.

VI- C- 3- Classe C

Dans cette classe l'adresse du réseau est codifiée sur 21 bits et l'adresse hôte sur 8 bits

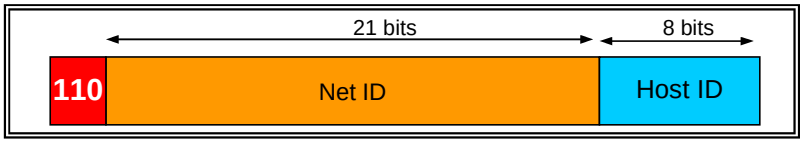


Figure VI-4 : Adressage IP Classe C.

Si les réseaux doivent être connectés à Internet les adresses des réseaux IP sont fournies par L'**InterNIC**[11] ou son représentant en France, le NIC France.

Pour faciliter le routage les adresses IP de classe C correspondent à des emplacements géographiques :

Adresses	Zone géographique
192.0.0 à 193.255.255	Adresses allouées avant la répartition géographique. Elles correspondent donc à plusieurs régions.
194.0.0 à 195.255.255	Europe
198.0.0. à 199.255.255	USA
200.0.0 à 201.255.255	Amériques centrale et du Sud
202.0.0 à 203.255.255	Pacifique

Vos notes :

_____

_____

_____

_____

_____

_____

_____

_____

[11] InterNic = International Network Information Center.

VI- C- 4- Classe D

Dans cette classe l'adresse du réseau est codifiée sur 28 bits et sert à diffuser des trames vers des groupes de stations.

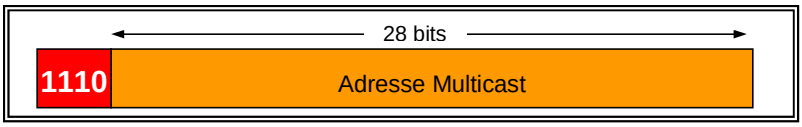


Figure VI-5 : Adressage IP classe D

VI- C- 5- Classe E

Cette classe est réservée à un usage futur.

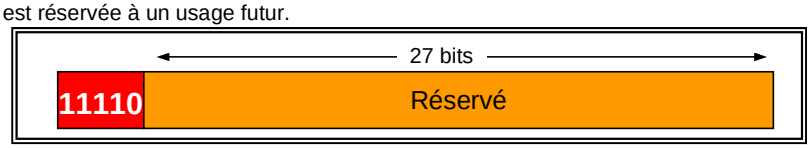


Figure VI-6 : Adressage IP Classe E.

VI- C- 6- Identification des classes d'adresses

Selon la valeur des bits du premier octet représentant l'adresse réseau IP, il est facile de déterminer la classe utilisée.

Classe	Gamme en notation décimale	Premier octet en binaire	Nb de réseaux	NB de noeuds
A	0.0.0.0 à **127**.255.255.255	**0** 0000000 et **0** 1111111	**126**	**16 777 214**
B	**128**.0.0.0 à **191**.255.255.255	**10** 000000 et **10** 111111	**16383**	**65534**
C	**192**.0.0.0 à **223**.255.255.255	**110** 00000 et **110** 11111	**2 097 151**	**254**
D	**224**.0.0.0 à **239**.255.255.255	**1110** 0000 et **1110** 1111		
E	**240**.0.0.0 à **247**.255.255.255	**11110** 000 et **11110** 111		

Figure VI-7 : Gammes d'adresses IP en fonction des classes.

VI- C- 7- Adresses Privées

Pour les réseaux non connectés à l'Internet, les administrateurs décident de la classe et de l'adresse NetID. Cependant pour des évolutions possibles, il est fortement recommandé de servir des adresses non utilisées sur Internet. Ce sont les adresses **privées** suivantes en classe A, B et C :

Tranches d'adresses IP privées	Nombre de réseaux privés
10.0.0.0 à 10.255.255.255	1 réseau de classe A
172.16.0.0 à 172.31.255.255	16 réseaux de classe B
192.168.0.0 à 192.168.255.255.	256 réseaux de classe C

Figure VI-8 : Adresses privées

VI- C- 8- Adresses spéciales

Les règles concernant les adresses IP prévoient un certain nombre d'adresses spéciales :

✓ Adresses **Réseaux** : Dans ces adresses, la partie réservée à l'adresse station est à 0. Par exemple, 126.0.0.0 représente l'adresse réseau et non l'adresse d'un hôte.

✓ Adresses **Broadcast** à diffusion dirigée : Dans ces adresses, la partie "adresse Station" a tous ses bits à **1**. Par exemple, 126.255.255.255 est une adresse de broadcast sur le réseau 126. Les routeurs peuvent transmettre cette trame vers le réseau 126.

✓ Adresses **Broadcast** à diffusion limitée. Dans ces adresses tous les bits sont à 1. (255.255.255.255) à. Cette trame est limitée au réseau de l'hôte qui l'envoie.

✓ Adresses pour la **maintenance** ou adresses "Loopback" : 127.0.0.1 (Ping sur la station pour vérifier le fonctionnement de la pile IP locale).

✓ Adresses **réservées** : Ce sont les adresses dont le numéro de réseau n'est composé que de 0 ou de 1.

VII - Réseaux et sous-réseaux

Un réseau peut être divisé en sous-réseaux afin de pouvoir :

- éviter le gaspillage des adresses nœuds d'un réseau
- utiliser des supports physiques différents.
- réduire le trafic sur le réseau.
- isoler une partie du réseau en cas de défaillance d'un composant du réseau.
- augmenter la sécurité.

Chaque sous-réseau est relié à un autre par un routeur.
Exemple :

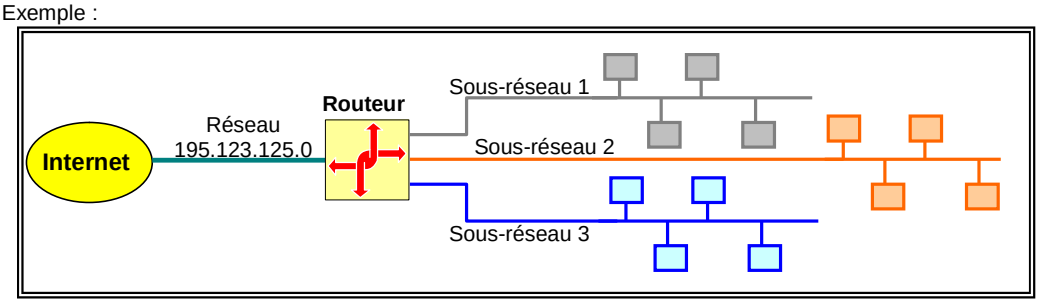


Figure VII-1 : Sous-réseaux.

Dans la figure ci-dessus, le routeur est connecté à Internet par un réseau de classe **C** 195.123.125.0. Il est donc possible d'utiliser 256 (- 2) adresses pour les nœuds. Cependant si tous les nœuds sont sur le même réseau, celui-ci risque d'être chargé. On répartit les nœuds sur 3 réseaux que l'on connecte à un routeur. Chacun de ces réseaux devant avoir une adresse distincte, on crée des adresses de sous-réseaux pour chacun d'eux.

VII-A- Masques de sous-réseaux

La notion de sous-réseaux était inexistante au début de IP. Elle est apparue avec la RFC 950 vers 1985. L'adressage de sous-réseaux va se faire avec des bits normalement réservés à l'adressage des nœuds.

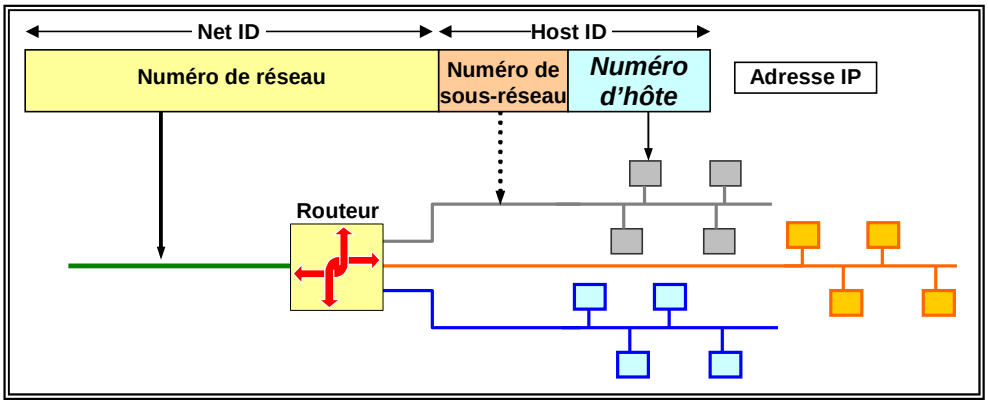


Figure VII-2 : Numérotation des sous-réseaux.

Pour indiquer le nombre de bits pris sur la partie HostID comme numéro de sous-réseau, on va utiliser un masque de sous-réseaux. Ce masque indique par des **bits à 1** le nombre de bits de l'adresse IP qui correspondent à l'adresse réseau et à l'adresse sous-réseaux. Les **bits à 0** du masque indiquent les bits de l'adresse IP qui correspondent à l'HostID.

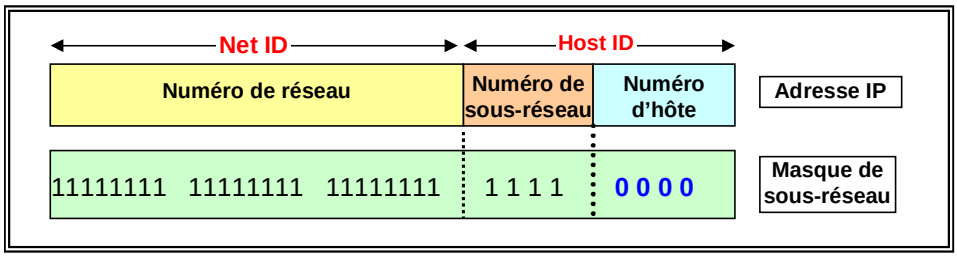


Figure VII-3 : Masque de sous-réseau.

Dans l'exemple ci-dessus, l'adresse IP est une adresse de classe C. On désire créer 16 sous-réseaux. Il est donc nécessaire d'utiliser 4 bits de la partie HostID pour indiquer le numéro de sous-réseau.
Le masque comporte **28** bits à **1**, c'est à dire :

- **24** bits correspondant à la partie NetID de l'adresse et **4** bits pour indiquer les bits de l'adresse IP qui doivent être interprétés comme étant l'adresse de sous-réseaux.
- **4** bits à **0**, indiquent les bits de l'adresse IP qui doivent être interprétés comme des adresses de nœuds.

Les masques de sous réseaux sont à entrer dans chaque ordinateur travaillant en IP. Les valeurs des masques se rentrent la plupart du temps en notation décimale pointée. Pour illustrer l'exemple ci-dessus, voici comment il conviendrait d'indiquer à une station NT, son adresse IP et son masque de sous-réseau.

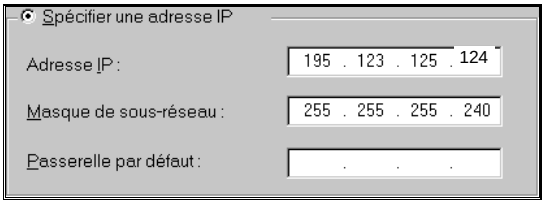


Figure VII-4 : Entrées de l'adresse IP et du masque de sous-réseau.

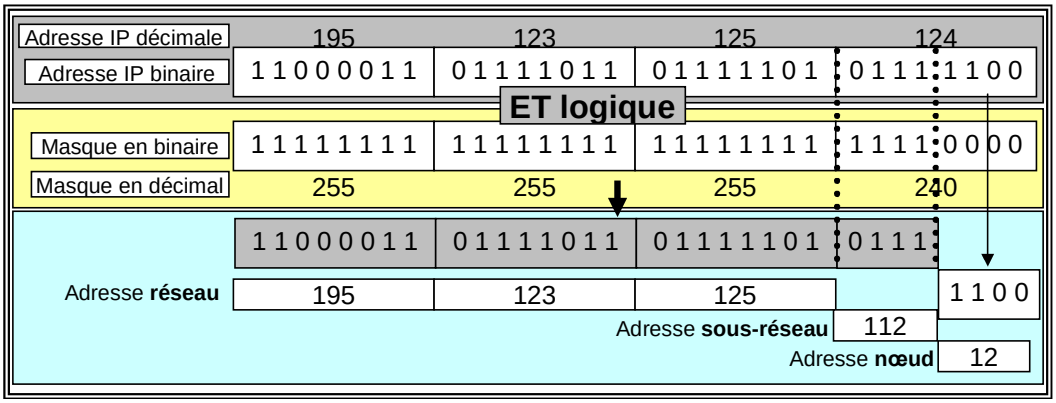


Figure VII-5 : Calcul de l'adresse de sous-réseau et de l'adresse nœud.

Dans cet exemple, le masque de sous-réseau comporte **28 bits**. L'adresse IP 195.123.125.124 est une adresse de classe C.

Les **24 premiers bits** du masque correspondent au **NetID**.

Les **4 bits** suivants à **1** dans le masque indiquent qu'il faut interpréter les 4 premiers bits du dernier octet comme une **adresse de sous-réseau** et non comme une adresse HostID. Les 4 bits à 0 du masque indiquent qu'il faut interpréter les 4 derniers bits du dernier octet de l'adresse IP comme une adresse nœud.

On calcule l'adresse du sous-réseau en tenant compte du poids binaire de chaque bit. Ici, (128 x 0) + (1 x 64) + (1 x 32) + (1 x 16)= 112. L'adresse nœud correspond aux 4 derniers bits de l'adresse IP (12).

VII-B- Exemple en classe B

	Masque-->	11111111 . 11111111 .	1111	*0000 . 00000000*	ID ss-réseau	ID nœud
Adresses IP	129.047.192.254	10000001 . 00101111 .	1100	*0000 . 11111110*		
	129.047.193.001	10000001 . 00101111 .	1100	*0001 . 00000001*		
	129.047.192.001	10000001 . 00101111 .	1100	*0000 . 00000001*		
	129.047.128.001	10000001 . 00101111 .	1000	*0000 . 00000001*		
	129.047.129.001	10000001 . 00101111 .	1000	*0001 . 00000001*		
	129.047.128.254	10000001 . 00101111 .	1000	*0000 . 11111110*		

Figure VII-6 : Exemple de masque de sous-réseaux avec des adresses de classe B.

Quelle est la valeur du masque en décimal. ? ……..
Quel est pour toutes les adresses IP le numéro de réseau ? ……..
Complétez le tableau en rentrant les adresses de sous-réseaux et les adresses nœud pour chaque adresse IP .

VII-C- Exemple en classe C

Un masque de sous-réseau est **255.255.255.240**
Quelle est la classe des adresses IP qui sont utilisées dans le tableau ci-dessous ? .....
En face des adresses IP suivantes, indiquez l'ID sous-réseau puis l'ID nœud :

Adresse IP	ID sous-réseau	ID nœud
195.252.13.33		
195.252.13.66		
195.252.13.47		

Figure VII-7 : Exemple de masque de sous-réseau en classe C.

Vos notes :

Mise en forme : Puces et numéros

~~V-~~VIII - Les services d'application utilisant TCP

Des applications réseau sont directement liées à la pile de protocole TCP/IP et la version "client" de ces applications est souvent livrée avec la pile de protocoles. Ces applications permettent par exemple le transfert de fichier (FTP), l'émulation de terminal en réseau (Telnet), l'affichage de page HTML (serveur et navigateur WEB), fonctions de nommage (DNS), …
Suivant les cas et le besoin de fiabilité des applications, elles utilisent soit le protocole **TCP,** soit le protocole **UDP** comme protocole de transport.

VIII-A- Ping

Ping (Packet Internet Grouper) est une application qui permet de vérifier le bon fonctionnement des composants d'un réseau utilisant TCP/IP. Elle permet par essais successifs de tester la pile IP et l'interface réseau de l'ordinateur sur lequel on se trouve, puis de tester les liaisons avec les autres machines du réseau. Cette application utilise le protocole ICMP véhiculé par IP.

```
# ping -c5 -s1000 126.0.0.1
PING 126.0.0.1 (126.0.0.1): 1000 data bytes
1008 bytes from 126.0.0.1: icmp_seq=0 ttl=255 time=14 ms
1008 bytes from 126.0.0.1: icmp_seq=1 ttl=255 time=14 ms
1008 bytes from 126.0.0.1: icmp_seq=2 ttl=255 time=14 ms
1008 bytes from 126.0.0.1: icmp_seq=3 ttl=255 time=14 ms
1008 bytes from 126.0.0.1: icmp_seq=4 ttl=255 time=14 ms
----126.0.0.1 PING Statistics----
5 packets transmitted, 5 packets received, 0% packet loss
round-trip (ms)  min/avg/max = 14/14/14 ms
```

Commande Ping avec paramètres demandant 5 envois de trames d'une longueur de 1000 octets. vers la machine dont @IP est 126.0.0.1

Statistiques de fin. Les trames ont mis en moyenne 14 ms pour faire l'aller et retour entre les 2 machines. Tous les paquets ont été retransmis.

Figure VIII-1 : Commande Ping.

~~V-1-~~VIII-B- FTP

Mise en forme : Puces et numéros

FTP (port 21) est une application qui permet d'assurer le transfert de fichiers, sans erreur, entre un micro-ordinateur et un hôte ou entre 2 hôtes. Un certain nombre de commandes propres à cette application permettent des transferts uniques ou multiples de fichiers dans les 2 sens (à l'alternat).

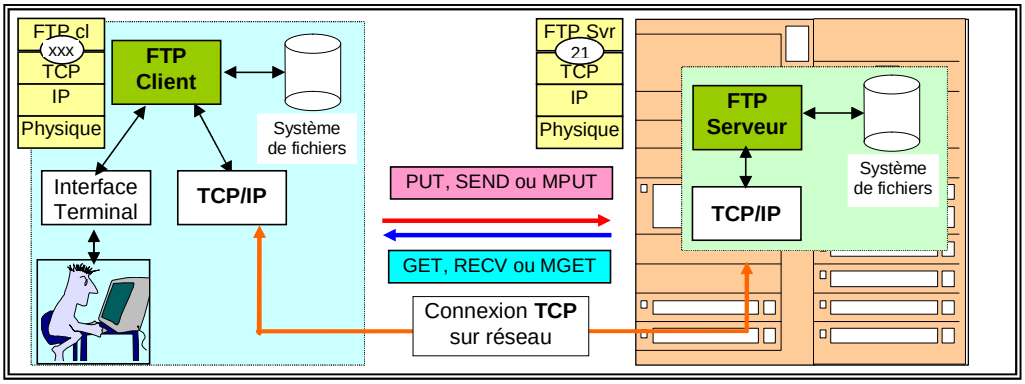


Figure VIII-2 : FTP.

Il existe une application **FTP en mode client** côté utilisateur. Dans l'autre ordinateur, une application **FTP serveur** tourne en tâche de fond.

```
# ftp
ftp> open gemini
Connected to gemini.
220 gemini FTP server (OSF/1 Version 5.60) ready.
Name (gemini:guest): guest
331 Password required for guest.
Password:
230 User guest logged in.
Remote system type is UNIX.
Using binary mode to transfer files.
```

Lance le FTP client (local)

Lance le FTP serveur (distant) et ouvre une connexion avec l'hôte Gemini.

Entrée du nom de connexion et du mot de passe

Transfert d'un fichier vers l'hôte.

```
ftp> put thelp6.txt
```

Le système renvoie un message, précédé de son code (150), suivi du nom du fichier, de l'adresse IP destination et du port FTP.

```
150 Opening BINARY mode data connection for thelp6.txt (126.0.2.1,1707).
netout: write returned 0?
226 Transfer complete.
ftp> close
221 Goodbye.
ftp> quit
# _
```

Le message de code 226 indique que le transfert est terminé

Cette commande met fin à la connexion avec gemini

Cette commande met fin au FTP local

Prompt Unix

Figure VIII-3 : Commandes FTP.

~~V-2-~~ VIII-C- Telnet

Mise en forme : Puces et numéros

Telnet (port 23) est un protocole qui permet **d'émuler** à partir d'un **micro-ordinateur** un **Terminal** connecté sur un Hôte à travers le **réseau**. Le type de terminal émulé peut être choisi dans une liste qui comporte toujours les terminaux VT100 ou VT220. Il existe de nombreuses versions de Telnet. Le Telnet **client** peut tourner sous système d'exploitation DOS, Windows ou Unix. Le Telnet **Serveur** (Telnet deamon = **telnetd)** tourne en général sous Unix. Au fur et à mesure que l'utilisateur tape les caractères au clavier, ils sont reçus par le serveur Telnet et transmis à Unix comme s'il s'agissait d'un terminal local en mode asynchrone connecté par une liaison série.

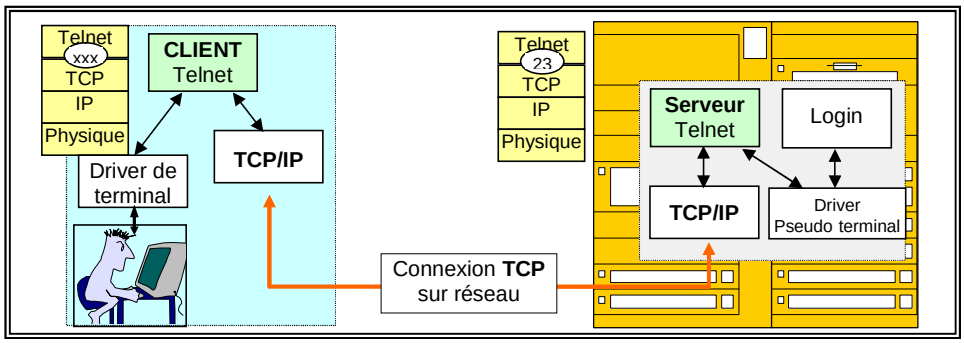


Figure VIII-4 : Telnet.

```
Last login: Fri Mar 14 15:22:59 from 126.0.1.121

Digital UNIX V4.0 (Rev. 386); Fri Nov  8 11:42:05 GMT+0100 1996

The installation software has successfully installed your system.

There are logfiles that contain a record of your installation.
These are:

        /var/adm/smlogs/install.cdf      - configuration description file
        /var/adm/smlogs/install.log      - general log file
        /var/adm/smlogs/install.FS.log   - file system creation logs
        /var/adm/smlogs/setld.log        - log for the setld(8) utility
        /var/adm/smlogs/fverify.log      - verification log file


% ls
backup          lprm.exe        mailer.exe      presentr.exe    resolvip.exe
bin             lwpcfg.exe      novasync.exe    prtgui.exe      user04
lpq.exe         lwpcfgd.exe     nvtsvc.exe      rapfiler.exe    windows
lpq000.exe      lwpcon.exe      nwunpack.exe    rarpd.exe
lpr.exe         lwptimer.exe    ping.exe        rcp.exe
```

Figure VIII-5 : Connexion à un serveur Telnet sous Unix .

~~V-3-~~ VIII-D- Les commandes R* d'Unix Berkeley

Mise en forme : Puces et numéros

Unix standard intègre dans ses commandes un ensemble appelé **commandes r**. Ces commandes permettent à partir d'un terminal connecté à un ordinateur sous Unix d'effectuer des opérations sur un autre ordinateur Unix du réseau.

Commande	Fonction
rlogin	Permet une connexion sur l'ordinateur spécifié.
rexec	Permet d'exécuter une commande sur l'ordinateur distant
rsh	Permet d'exécuter un shell sur un ordinateur distant.
rcp	Permet la copie de fichiers entre machines distantes.
rwho	Affiche les utilisateurs connectés sur le réseau.

Le protocole TCP-IP

rwall	Transmet un message aux utilisateurs connectés aux machines indiquées.
ruptime	Affiche des informations sur les ordinateurs du réseau.

Figure VIII-6 : Commandes r*

~~V-4-~~VIII-E- WWW

Mise en forme : Puces et numéros

Le Wide World Web est l'ensemble des serveurs qui stockent des documents au format **HTML**[12] (et autres) sur Internet. Pour assurer le dialogue entre les clients Web (les navigateurs Web) et les serveurs, on utilise le protocole **HTTP**[13] qui s'appuie sur TCP et IP.

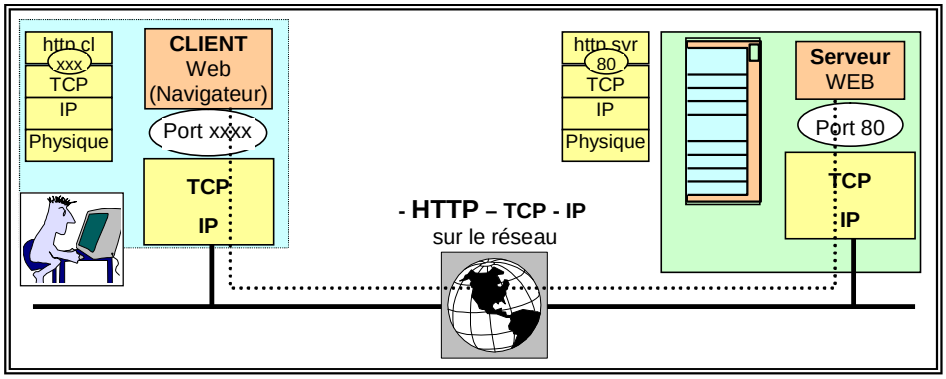


Figure VIII-7 : Liaison WWW.

~~V-7-~~VIII-F- Les protocoles de messagerie SMTP, POP et IMAP4

Mise en forme : Puces et numéros

La messagerie est un des services d'Internet les plus utilisés et les plus pratiques. Plusieurs protocoles sont utilisés pour la transmission des messages.

VIII- F- 1- SMTP

Le protocole SMTP[14] permet d'envoyer les messages en ASCII vers le serveur du provider[15] auquel on est raccordé

Lorsque l'utilisateur A veut envoyer un message, il le compose tout d'abord en utilisant un utilitaire de messager, (Outlook, Outlook Express, Message Composer, Eudora ou autre). Le message composé est d'abord envoyé vers une boîte d'envoi locale. Puis, le message est acheminé vers le serveur du provider à l'aide du protocole SMTP. Pour envoyer des messages non textuels (images, documents Word, programmes), on est obligé d'utiliser des utilitaires pour rendre la transmission compatible avec SMTP qui ne supporte que l'ASCII. Ces utilitaires sont UUENCODE / UUDECODE ou plus récemment MIME[16].

[12] HTML = Hypertext Markup Language
[13] HTTP = Hypertext Transfer Protocol.
[14] SMTP= Simple Mail Transfert Protocol.
[15] Provider= Fournisseur de services Internet.
[16] MIME= Multipurpose Internet Mail Extensions.

VIII- F- 2- POP et IMAP4

Le message est ensuite acheminé vers le serveur sur lequel est connecté le destinataire. Celui-ci n'étant pas forcément relié en permanence à ce serveur, il existe une boîte aux lettres personnelle dans laquelle seront stockés tous les messages non lus.
Le destinataire consulte sa boîte aux lettres et récupère ses messages grâce au protocole POP3[17]. Un protocole plus récent IMAP4[18] est parfois utilisé. Il possède l'avantage de permettre la consultation des messages sur le serveur sans forcement les rapatrier dans la station.

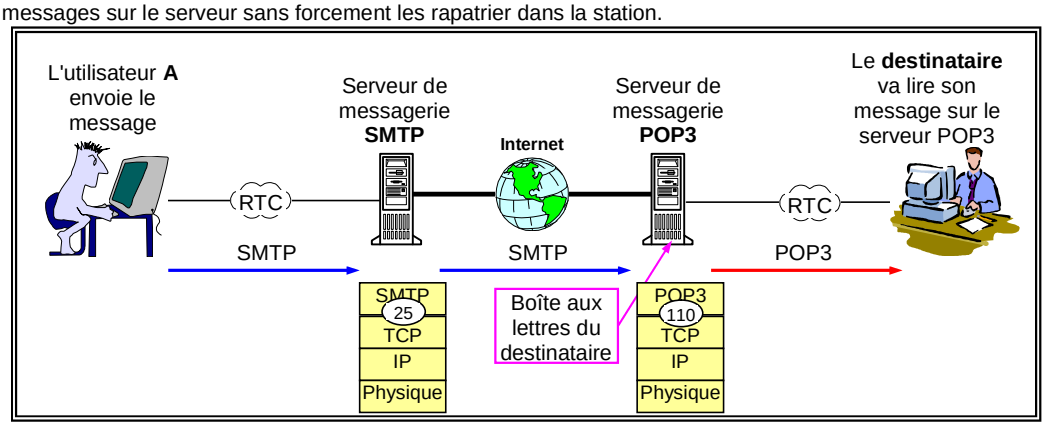


Figure VIII-8 : Messagerie Internet.

Vos notes :

[17] POP3= Post Office Protocol Version 3.
[18] IMAP4= Internet Message Access Protocol Rev4.

IX - Services d'applications utilisant UDP

~~V-5-~~ IX-A- DNS

Mise en forme : Puces et numéros

DNS[19] est un service qui permet sur un réseau IP et plus particulièrement sur Internet de résoudre le problème de nommage des ordinateurs. En effet, il est plus facile pour l'utilisateur d'utiliser pour se connecter à un serveur Web par exemple, une adresse du type *www.amora.fr*, que de taper une adresse IP difficile de mémoriser. Le rôle de DNS est donc de faire une équivalence entre un nom de machine et son adresse IP.

Pour ce faire, on utilise :

- un système de nommage des ordinateurs qui est normalisé et hiérarchisé de manière à ce que chaque ordinateur de l'Internet porte un nom unique.
- des serveurs de noms DNS qui comportent dans des bases de données le nom des ordinateurs et l'adresse IP correspondante.

IX- A- 1- Système de nommage hiérarchisé

L'espace de noms du DNS est organisé d'une façon hiérarchique à la manière d'un système de fichiers DOS ou UNIX.

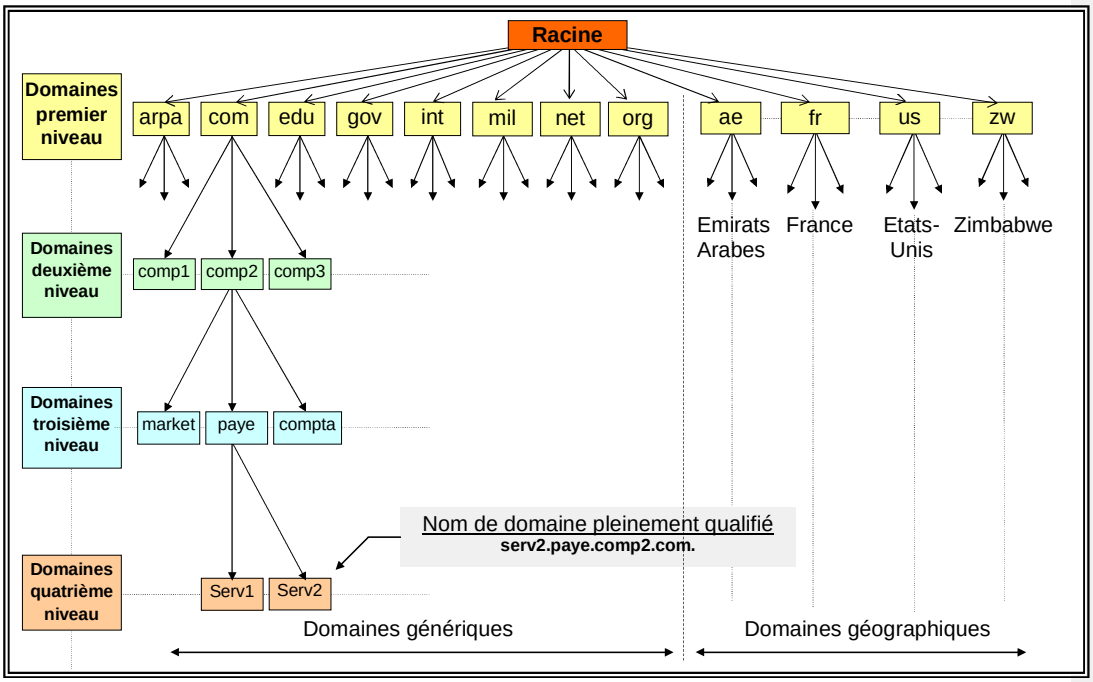


Figure IX-1 : DNS.

L'arbre démarre par une racine puis, à chaque niveau, on trouve des nœuds qui permettent l'apparition de nouvelles branches de l'organisation hiérarchique. Chaque nœud de la figure représente un **domaine** qui possède un **label** qui peut comporter jusqu'à 63 caractères.

Le **nom de domaine** d'un nœud quelconque de l'arbre est la **liste des labels** permettant d'atteindre la racine. On commence par écrire le label de plus bas niveau et on termine par celui du plus haut niveau. Chaque label est séparé du suivant par un **point**.

Si le nom de domaine se termine par un point, c'est un **nom de domaine absolu** ou **nom de domaine pleinement qualifié**. (FQDN= Fully Qualified Domain Name).

Un nom de domaine doit être unique, mais il peut exister des labels identiques à des niveaux différents. Les domaines de niveau supérieur **com** (commercial), **edu** (éducation), **gov** (gouvernemental), **int** (international), **mil**, (militaire), **net** (réseau) et **org** (autres organisations) sont appelés **domaines génériques**. Ceux qui au même niveau représentent un nom de pays (fr, us, ...) sont appelés **domaines géographiques**.

[19] DNS = Domain Name System.

IX- A- 2- Serveurs de noms de domaine

Une **zone** est une partie de l'arbre DNS administrée séparément. Chaque zone doit posséder des **serveurs de noms**. Pour chaque machine de la zone, l'administrateur doit entrer dans le serveur de noms, l'adresse IP de cette machine et le nom de domaine. Si un serveur de noms ne contient pas le nom demandé par un ordinateur, il doit être capable d'interroger les autres serveurs de noms des niveaux supérieurs. Tout nom demandé et trouvé par un serveur de noms est mis en mémoire cache. Ceci évite des demandes répétées.
Pour le réseau mondial **INTERNET**, le **NISC**[20] attribue des *noms de domaines* en respectant des conventions propres à cet organisme. Chaque société ou chaque université peut recevoir un nom de domaine de haut niveau (*com* pour une société commerciale et *edu* pour un établissement d'éducation par exemple) et un nom de sous domaine propre à la société ou à l'université (par exemple *novell* ou *ucla*).
A partir de ce point l'administrateur du domaine peut continuer la structure hiérarchique et prolonger le nom de sous-domaine qui peut devenir par exemple : *messagerie.service_technique.novell.com* pour une société commerciale ou *sectionA.lettres.ucla.edu* pour un département d'une université. Le niveau le plus bas de la hiérarchie apparaît au début du nom de domaine.
L'implémentation de DNS utilisée sur les ordinateurs Unix est appelée **BIND**[21] et le deamon serveur *named.*
Les serveurs DNS peuvent aussi être implémentés sur système d'exploitation NetWare ou Windows NT 4 Server.

IX- A- 3- Exemple d'utilisation de DNS

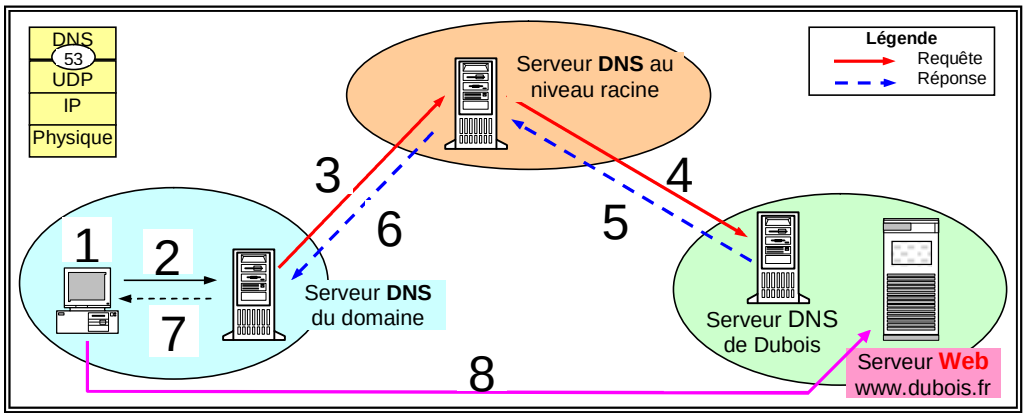


Figure IX-2 : Fonctionnement DNS.

-1- L'utilisateur tape l'URL dans le navigateur Web de sa station http://www.dubois.fr. Un programme nommé résolveur de nom va interroger le serveur DNS de la zone de l'utilisateur pour connaître l'adresse IP correspondant à www.dubois.fr.
-2- Une requête est envoyée par le résolveur au serveur DNS. Ce serveur de noms de domaine ne connaît pas l'adresse IP. Il va donc interroger le serveur de nom du niveau racine.
-3- Le serveur DNS de la zone de l'utilisateur envoie une requête au serveur DNS du niveau racine.
-4- Ce serveur ne connaît pas non plus l'adresse IP. Il interroge le serveur DNS du domaine dubois.fr
-5- Ce serveur DNS renvoie une réponse qui contient l'adresse IP. Elle est mise en cache dans le serveur DNS du niveau racine.
-6- Ce serveur renvoie la réponse vers le serveur DNS demandeur. Celui-ci met l'adresse dans un cache pour éviter d'autres requêtes ultérieures
-7- L'adresse IP demandée est fournie au résolveur de la station utilisateur qui la met aussi en cache.
-8- La connexion est établie à travers le réseau Internet vers le serveur Web .

Mise en forme : Puces et numéros

[20] NISC = Network Information Systems Center
[21] BIND = Berkeley Internet Domain Name

~~V-6-~~ IX- A- 4- Désignation des serveurs DNS sur les stations IP

Dans les stations PC sous Windows 95/98 ou NT, pour utiliser le service DNS, il faut dans la configuration des propriétés de la pile IP, déclarer le nom de la station, le domaine où elle se trouve et l'adresse IP du ou des serveurs DNS qu'elle doit contacter pour la recherche des adresses IP.

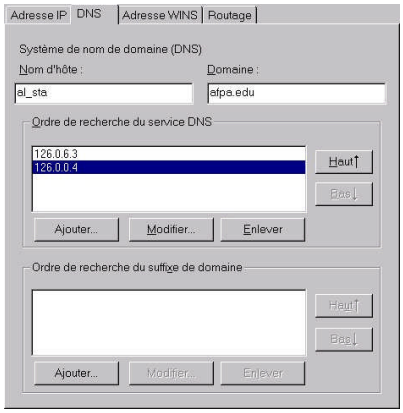


Figure IX-3 : Configuration de DNS sur une station NT ou 95.

IX-B- NFS

IX- B- 1- Principes

NFS (Network File System) est une application qui permet à une machine d'exporter son système de fichiers (tout ou partie) vers le réseau. La base de NFS a été développée par SUN Microsystems dans les années 80. Il est devenu un standard dans le monde Unix. Pour pouvoir avoir accès aux fichiers exportés à partir d'un autre ordinateur, il faut qu'il soit équiper d'un **client NFS**.
La machine qui exporte ses fichiers est souvent un ordinateur sous Unix, mais les clients peuvent être des machines Unix ou Macintosh ou des PC sous DOS, Windows 9x ou NT.

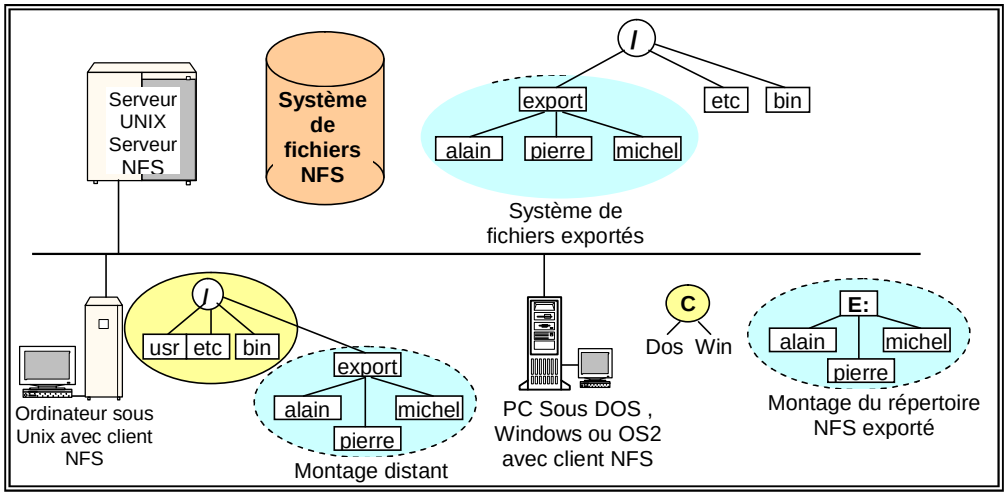


Figure IX-4 : Exportation NFS.

La figure ci-dessus montre un répertoire *export* exporté par le serveur NFS d'un ordinateur tournant sous UNIX.
L'ordinateur client UNIX voit ce répertoire comme une extension de son propre système de fichiers UNIX.
La station PC monte le répertoire exporté pour le faire apparaître comme une unité de logique DOS-Windows E:.

Mise en forme : Puces et numéros

~~VI-~~IX-C- TFTP

TFTP[22] est un protocole qui est une version allégée de FTP. Il utilise UDP comme protocole de transport. Celui-ci n'étant pas fiable, TFTP se sert de son propre système d'accusé de réception pour assurer une bonne qualité de transmission.
TFTP est utilisé principalement pour charger à partir d'un serveur TFTP, le système d'exploitation d'ordinateurs sans disque ou de terminaux X-Windows par exemple. Il est aussi utilisé pour assurer le mise à jour des OS des matériels réseaux (hubs, ponts, commutateurs, routeurs) contenus dans des mémoires non volatiles, mais réinscriptibles.

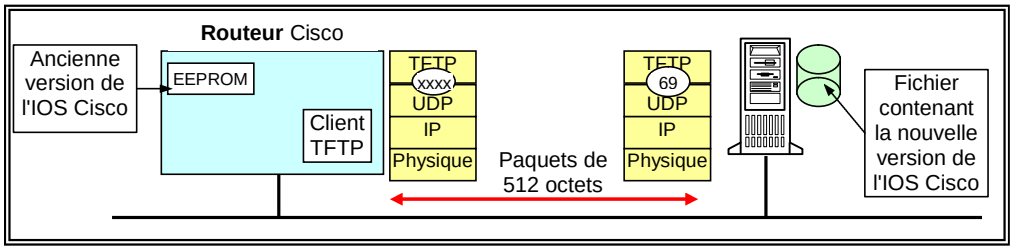


Figure IX-5 : TFTP.

Le logiciel TFTP client est souvent implanté en ROM. Sa simplicité permet d'obtenir une taille de programme compatible avec la capacité de stockage des ROMs.

IX-D- SNMP

SNMP[23] est un protocole qui permet la gestion cohérence d'un inter-réseau et la surveillance des éléments du réseau. Les éléments du réseau sont les stations, les serveurs, les routeurs, les hubs, etc.
La gestion du réseau repose sur 3 entités :

- le ou les **managers SNMP** qui sont installés dans une ou plusieurs **NMS** (Network Management Station) et les **agents SNMP** qui sont installés dans les éléments du réseau à gérer.
- une base de données informationnelle de gestion (**MIB**[24]) qui définit les variables utilisées dans chaque élément du réseau. Chaque **variable** est repérée par un **Objet identifier** unique.
- les différents types d'objets (Counter , String, ...)

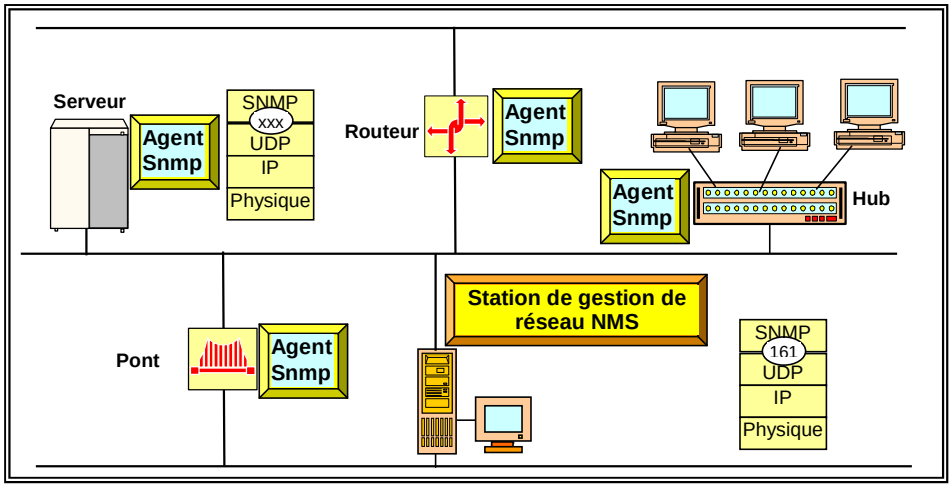


Figure IX-6 : Agents et Manager SNMP.

[22] TFTP= Trivial File Transfert Protocol.
[23] SNMP=Simple Network Management Protocol
[24] MIB : Management Information Base.

A partir de la station de gestion, l'administrateur du réseau peut *consulter ou modifier la configuration* des éléments du réseau. Cette fonction est réalisée grâce à une procédure du type requête-réponse.

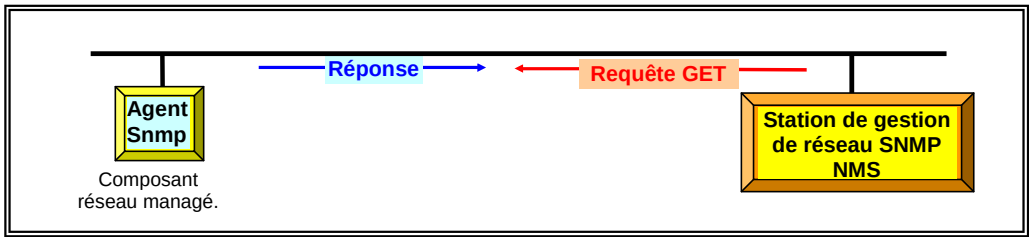


Figure IX-7 : Requête – Réponse SNMP.

Les agents sont par ailleurs capables d'envoyer des messages (**Traps**) pour prévenir l'administrateur des problèmes survenant sur les composants du réseau.

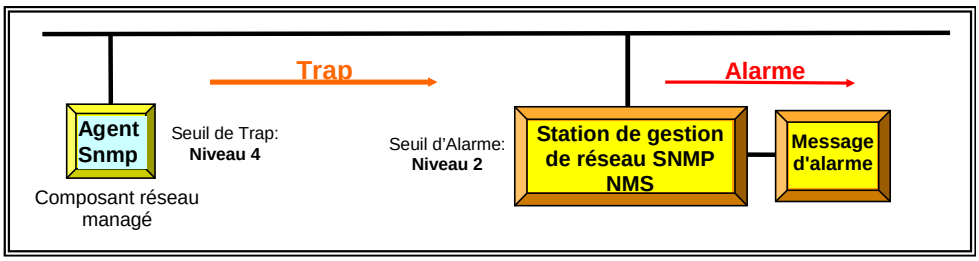


Figure IX-8 : Traps SNMP.

Les traps sont envoyés vers la station de management ou dans des trames broadcast à partir d'un certain seuil de gravité déterminé par l'administrateur (noté de 0 à 7).
Une fois arrivés sur la station de gestion, les traps sont enregistrés dans un fichier d'événements. Si leur niveau de gravité dépasse un certain seuil fixé au préalable, ils déclenchent une **alarme** qui peut être un signal sonore, un affichage sur l'écran, l'envoi d'un message sur le réseau ou sur un système de mobile (pager ou téléphone) ou même l'exécution de n'importe quel programme.

Vos notes personnelles :

X - Fichiers associés à TCP/IP

TCP/IP utilise 4 fichiers de base de données pour convertir des données internes, telles que les **adresses IP**, en **noms** plus faciles à utiliser. Le contenu de ces fichiers une fois lu, est placé en mémoire cache pour éviter des accès répétitifs aux disques. Ces fichiers, en mode texte, se trouvent en général dans le répertoire **\etc** des hôtes TCP/IP.

X-A- Fichier Hosts

Ce fichier, que l'on écrira avec un éditeur de texte simple, contient des entrées qui permettent de rendre une adresse IP équivalent à un ou plusieurs noms.
La syntaxe est la suivante :

adresse_IP nom_d'hôte [alias [...]]

L'*adresse_IP* est donnée en notation décimale point ou en valeur hexadécimale commençant par **0x**.
Le *nom_hôte* est le nom du système associé à l'adresse IP. Ce nom ne doit pas contenir d'espace et doit être unique.
L'*alias est un autre nom qui désigne le système ou une autre manière de l'écrire (Majuscules-minuscules ou abréviations).*
Exemple de fichier **\etc\hosts** : Les lignes précédées de # sont des commentaires.

```
#
# Mapping of host names and host aliases to IP Addresses
#
127.0.0.01    loopback  lb  localhost  # loopback address
#
# exemples d'adresses, de noms d'hôtes et d'alias
126.0.0.1      Mugix MUGIX mugix mu
126.0.0.2      Jarrix JARRIX jarrix ja
126.0.2.1      GEMINI gemini gem
126.0.2.2      APOLLO apollo  apo APO Apo
```

Figure X-1 : Exemple de fichiers **hosts**.

X-B- Fichier Networks

Le fichier \etc\networks contient les informations sur les réseaux de l'inter-réseau.
La syntaxe est la suivante :

nom_réseau numéro_réseau [/masque_réseau] [alias [...]]

Le *nom_réseau* est le nom du réseau. Il ne peut contenir d'espace, de tabulation ou le symbole #. Il doit être unique.
Le *numéro_réseau* est l'adresse IP donnée au réseau.
Le *masque_réseau* est le masque de sous-réseau du réseau. Ce champ est facultatif
L'*alias* est un autre nom donné au réseau. Il peut y avoir jusqu'à 10 alias pour le même réseau.
Exemple de fichier \etc\networks :

```
#
# Networks numbers
#
loopback 127   # réseau fictif interne pour bouclage
#
# réseaux de l'inter-réseaux
angers              126  ang      #réseau local afpa angers
greno               123  gre      #réseau local afpa grenoble
```

Figure X-2 : Exemple de fichiers networks.

X-C- Fichier Protocol

Le fichier \etc\protocol contient des informations sur les protocoles connus utilisés sur l'inter-réseau. Chaque ligne fournit des informations sur un protocole.
La syntaxe est la suivante :

nom_protocole numéro_protocole [alias [...]]

Le *nom_protocole* est le nom du protocole associé au numéro qui suit.
Le *numéro_protocole* est le numéro du protocole.
L'*alias* est un autre nom donné au protocole.
Exemple de fichier \etc\protocol :

```
# internet (IP) protocols
ip      0       IP      # protocol internet
icmp    1       ICMP    #protocol de message d'erreurs dans l'inter-réseau
igmp    2       IGMP    # protocol multicast
ggp     3       GGP     # gateway-gateway protocol
tcp     6       TCP     # protocol de transmission
udp     17      UDP     # user datagram protocol
```

Figure X-3 : Exemple de fichier \etc\protocol.

X-D- Fichier Services

Le fichier **\etc\services** contient des informations sur les services utilisés sur l'inter-réseau IP. La syntaxe est la suivante :

nom_service numéro_port/nom_protocole [alias [...]]

Le *nom_service* est le nom de service associé au port dont le nom ou le numéro suit. Ces services sont des services des couches session, présentation ou application, tels Telnet, FTP, TFTP ou SMTP.
Le *numéro_port* est le numéro de port utilisé par le service.
Le *nom_protocole* désigne le protocole auquel le service est lié. Il s'agit, en général, d'un protocole du niveau transport comme TCP ou UDP.
L'*alias* est un autre nom donné au service

Exemple de fichier \etc\service

```
                TCP Ports
# Service Name Port/Protocol Aliases

echo            7/tcp
discard         9/tcp     sink null
systat          11/tcp    users
daytime         13/tcp
netstat         15/tcp
qotd            17/tcp    quote
chargen         19/tcp   ttytst source
ftp-data        20/tcp
ftp             21/tcp
telnet          23/tcp
smtp            25/tcp     mail
time            37/tcp     timserver
name            42/tcp     nameserver
whois           43/tcp     nicname
nameserver      53/tcp    domain
apts            57/tcp
apfs            59/tcp
rje             77/tcp         netrjs
finger          79/tcp
link           87/tcp         ttylink
hostnames      101/tcp        hostname
iso-tsap       102/tcp        tsap
x400           103/tcp
x400-snd       104/tcp
pop-2          109/tcp  pop postoffice
```

```
                #UDP ports
# Service Name Port/Protocol  Aliases

echo            7/udp
discard         9/udp     sink null
systat          11/udp    users
daytime         13/udp
netstat         15/udp
qotd            17/udp     quote
chargen         19/udp     ttytst
source
time            37/udp     timserver
rlp             39/udp     resource
name            42/udp     nameserver
whois           43/udp      nicname
nameserver      53/udp      domain
bootps          67/udp      bootp
bootpc          68/udp
tftp            69/udp
sunrpc          111/udp
erpc            121/udp
ntp             123/udp
statsrv         133/udp
profile         136/udp
snmp            161/udp
snmp-trap       162/udp
at-echo         204/udp
```

Figure X-4 : Ports TCP et UDP.

XI - Résolution de noms d'hôtes IP

Problème : Comment résoudre la correspondance entre les noms des ordinateurs en réseau et leur adresse IP ?

XI-A- Les noms NetBIOS

Ce sont des noms utilisés sur le réseau pour identifier à l'aide de NetBIOS les ordinateurs connectés au réseau Microsoft. Ces noms d'ordinateurs sont entrés au moment de l'installation de Windows for Workgroup, Windows 95(98) ou Windows NT. Ces noms doivent être uniques sur le réseau (ce qui pose des problèmes sur les grands réseaux). Ils ne doivent pas dépasser 15 caractères et ne pas comporter certains caractères.
Les noms NetBIOS peuvent être modifiés dans le panneau de configuration.
Ces noms NetBIOS sont particulièrement utilisés dans la recherche des ressources dans les réseaux Microsoft en utilisant la convention **UNC** (Universal Naming Convention).

Par exemple, la recherche d'un fichier sur un réseau utilise une UNC composée de 3 parties :
- Un nom d'ordinateur NetBIOS précédé de \\
- Un **nom de partage** situé sur l'ordinateur (option)
- Un **nom de répertoire** ou de **fichier** au format MS-DOS situé dans le partage.

Exemple : **\\berlioz\serv_nt4**

XI-B- II- Noms de domaines

XI- B- 1- Domaines Windows NT

Les **Domaines Windows NT** correspondent à un ensemble d'ordinateurs en réseau pour lesquels une base de données d'authentification a été créée (**SAM** = Security Account Manager). Ce concept est spécifique à Windows NT. Les noms de domaine sont créés au moment de l'installation du premier serveur NT dans un domaine. Ce serveur est nommé Contrôleur Principal de Domaine (CPD).

XI- B- 2- Domaines Internet

Les **Domaines DNS** sont des zones dans une structure hiérarchisée de l'Internet qui correspondent à des serveurs gérés par le même administrateur. Les serveurs de noms de Domaines DNS de l'Internet dialoguent entre eux et permettent de fournir à un client de l'Internet l'adresse IP de n'importe quel serveur du réseau.
Les noms de domaine se présentent sous la forme :

microsoft.com ou **ibm.com**

Les noms des serveurs de l'Internet sont associés aux noms de domaines et se présentent sous la forme :

www.microsoft.com **www.ibm.com**

XI-C- Correspondance entre les noms des ordinateurs et les adresses IP

XI- C- 1- Solution de départ, le fichier Hosts

Pour de petits réseaux non connectés sur l'Internet, la correspondance entre les noms des ordinateurs et les adresses IP peut être résolue localement grâce à une table consignée dans un fichier nommé **hosts** sur chaque ordinateur. Cette méthode ne peut plus être utilisée lorsque le nombre d'ordinateurs sur le réseau devient important, car tout ajout d'un nouvel ordinateur sur le réseau doit entraîner une mise à jour du fichier **hosts** dans chaque ordinateur. De plus, ce système ne résout pas les problèmes de correspondance entre les noms de serveurs sur Internet et leur adresse IP.

N.B. Le nom d'ordinateur donné dans le fichier **hosts** est purement local à l'ordinateur où il se trouve et ne correspond pas forcément au vrai nom de l'ordinateur.

XI- C- 2- Solution pour Internet, DNS

Domain Name System assure la correspondance entre le **nom Internet complet** d'un ordinateur (nom de serveur + nom de domaine) et son adresse IP.

Ravel.afpa.edu correspond à 126.0.6.3

Pour utiliser ce service sur une station de travail Win95 ou NT, dans la fenêtre TCP/IP, il faut sélectionner l'onglet **DNS** et indiquer l'adresse du serveur **DNS** primaire.
Dans chaque domaine **Internet**, il existe un ou plusieurs serveurs de Noms de Domaine. Ces serveurs contiennent une table de correspondance entre les noms des ordinateurs de la zone et leur adresse IP. Ces tables sont maintenues manuellement par l'administrateur de la zone.

XI- C- 3- Solutions sur un réseau avec NT

LMHOSTS
lmosts à l'instar du fichier hosts est un fichier qui contient une table de correspondance statique entre les noms NetBIOS des ordinateurs du réseau et leur adresse IP. Cette table doit être maintenue manuellement et ne peut donc être utilisée que sur de petits réseaux.
Cependant, sur certains OS, il est possible d'utiliser un fichier **lmhosts** partagé, se trouvant sur un autre ordinateur. Dans ce cas, il est possible de gérer ce fichier d'une manière centralisée.

WINS
Windows Internet Name Service utilise une couche NetBIOS au-dessus de TCP/IP pour établir des correspondances entre les adresses IP et les noms des ordinateurs NT ou 95.

Ravel correspond à **126.0.6.3**

Le service **WINS** tourne un ou plusieurs serveurs NT du réseau.
Les noms des ordinateurs et les adresses correspondantes sont contenus dans une base de données sur le serveur WINS. Cette base de données dynamique est mise à jour automatiquement sans intervention humaine (contrairement à DNS). Cette base de données est plate et ne possède donc pas de niveaux hiérarchiques à l'opposé de DNS.
Pour qu'une station client puisse utiliser le service de résolution d'adresses WINS, il faut dans la fenêtre TCP/IP utiliser l'onglet WINS et indiquer au moins l'adresse IP du serveur WINS (serveur primaire). S'il existe plus d'un serveur WINS, on peut entrer une deuxième adresse IP (serveur secondaire).
Si la case "**Activer la résolution DNS pour Windows**" est cochée, les noms d'ordinateurs NetBIOS sont d'abord recherchés dans le serveur DNS. En effet, il peut exister un lien entre les serveurs DNS et WINS. C'est-à-dire que si un nom d'ordinateur n'est pas trouvé dans le serveur **DNS**, il est recherché dans le serveur **WINS**.
Si la case "**Activer la recherche de clé LMHOSTS**", le système recherche des correspondances entre le nom d'ordinateurs et les adresses IP dans le fichier **lmhosts**.

XII - Protocole IP

XII-A- IP et les réseaux physiques

Le protocole IP permet d'avoir une vue globale de l'inter-réseau sans tenir compte des différents types de réseaux physiques qui le constituent. Les protocoles TCP et UDP ne dialoguent qu'avec IP sans voir les réseaux physiques. Les datagrammes IP peuvent être encapsulés dans des trames Ethernet, Token-Ring ou des paquets X25 ou même des liaisons séries asynchrones en utilisant un protocole de transports sur ce type de liaison comme PPP[25].
IP est aussi utilisable sur des liaisons spécialisées ou des réseaux de type Frame Relay ou ATM[26].

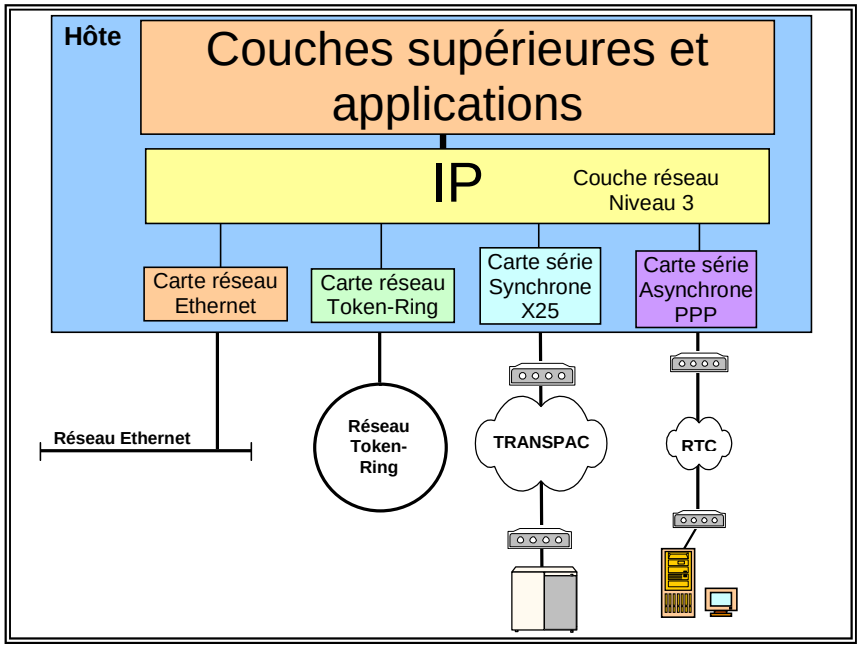


Figure XII-1 : IP et les réseaux physiques.

XII-B- Fragmentation

IP génère des paquets de taille variable en fonction du nombre d'octets de données à transporter et de la **MTU**[27] du réseau physique. La MTU d'un réseau Ethernet est de 1500 octets, alors que celle d'un réseau Token-Ring à 16 Mbps est de 17940 octets. Il se peut donc qu'un paquet IP encapsulé dans une trame Token-Ring soit **fragmenté** en plusieurs paquets IP pour être véhiculé ensuite dans une trame Ethernet ou des paquets X25.

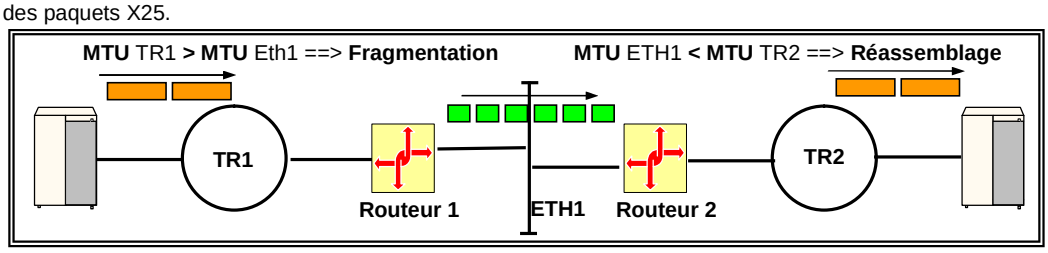


Figure XII-2 MTU, Fragmentation et Réassemblage.

[25] PPP= Point To Point Protocol.
[26] ATM = Asynchronous Transfert Mode
[27] MTU= Maximum Transmission Unit. Taille maximum d'une trame sur un réseau physique

XII-C- Datagramme

IP travaille en mode datagramme sans connexion, c'est-à-dire que chaque paquet IP est véhiculé dans un internet de manière indépendante. Il se peut donc que plusieurs paquets successifs empruntent des chemins différents.

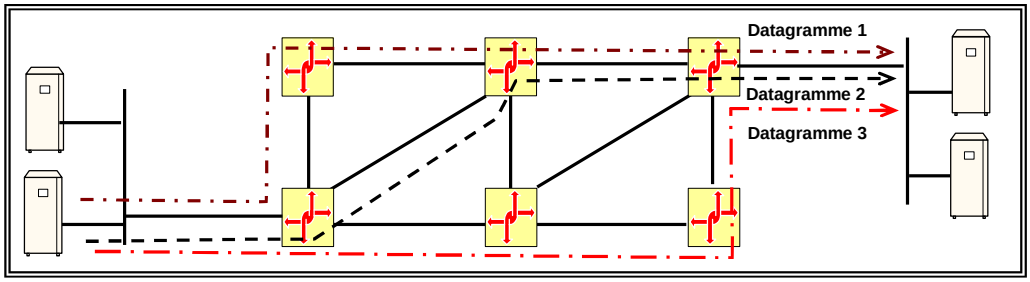


Figure XII-3 : Datagrammes IP.

L'ordre d'arrivée des datagrammes peut être différent de celui de départ. On dit que le protocole IP n'assure pas le **séquencement**, ni d'ailleurs la fiabilité de la transmission (pas **d'accusé de réception**, ni de **checksum** sur les données transportées). Ces fonctions, si elles sont nécessaires pour les applications qui utilisent IP doivent être assurées par le protocole de la couche supérieure à IP. **TCP** assure le séquencement des paquets, les accusés de réception et la checksum.

XII-D- Format de l'en-tête

Le paquet comporte un en-tête renfermant entre autres, les adresses IP Source et Destination. L'en-tête est suivi d'un champ comprenant les données envoyées par les couches supérieures de la pile. Ce paquet sera ensuite encapsulé dans une trame du réseau physique utilisé.

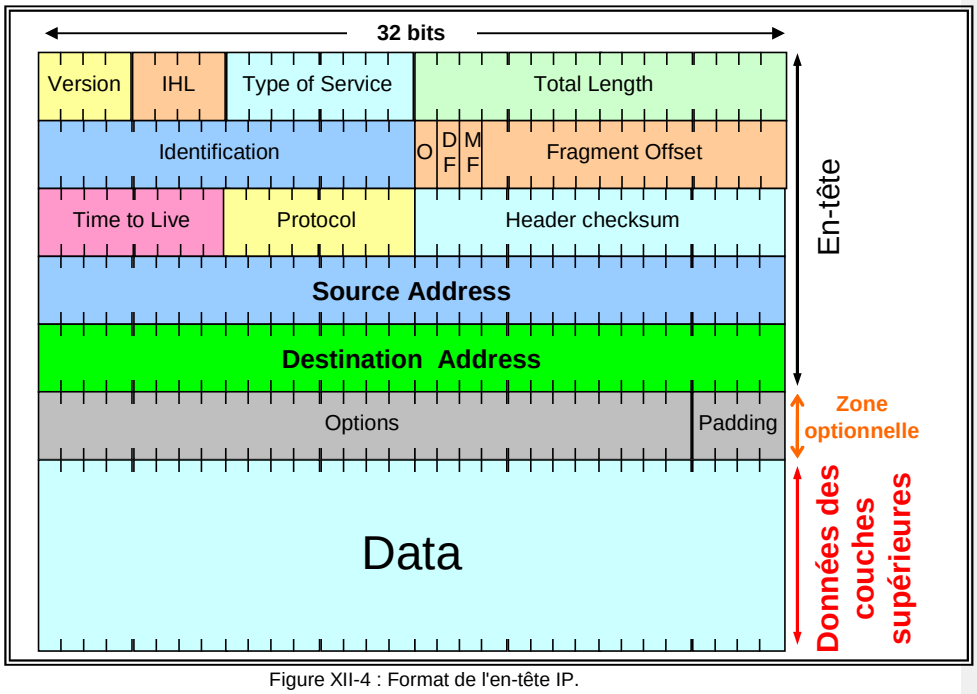


Figure XII-4 : Format de l'en-tête IP.

Octets	Nom du champ	Signification
1	Version	Ce champ est formé par les 4 bits de poids fort du premier octet. Il indique la version de IP utilisée. La valeur actuelle est 4 (O1OO). Elle passera à 6 lorsque la version IPV6 sera opérationnelle.
1	IHL	Internet Header Length = Longueur de l'en-tête IP. On y trouve une valeur qui indique la longueur de l'en-tête en multiple de 32 bits. Par exemple si la valeur est 0101 (5), cela veut dire que l'en-tête mesure 5 fois 32 bits, soit 20 octets. Ce champ est indispensable, car la longueur de l'en-tête varie s'il y a des options à la fin de l'en-tête standard.
2	TOS	TOS= Type of Service. Ce champ informe les routeurs des réseaux de la qualité de service désirée. Il est divisé en 6 parties. *1- 'Precedence'* : Les 2 bits de poids fort de l'octet indique la 'préséance', c'est à dire en fait l'importance du paquet IP et la priorité que l'on doit lui accordé. La valeur par défaut de ce champ peu utilisé est 0. 2- Délai : 0= normal, 1= Elevé. 3- Débit : 0= normal, 1= Elevé. 4- Fiabilité : 0 = Normale, 1= Elevée. 5- Coût: 0= Normal, 1= Elevé. 6- Ce bit doit être à 0. Le TOS est peu utilisé, mais il est fonctionnel. Les valeurs du TOS sont gérées par les couches supérieures.
3-4	Total Length	Ce champ indique la longueur totale du paquet IP.
5-6	Identification	Ce champ indique le numéro du paquet émis par la couche réseau d'un nœud. Le compteur compte de 0 à 65535, puis repasse à 0.
7-8	Fragment Offset	Si le bit DF (Don't Fragment) de ce champ est à 1, cela signifie que le datagramme IP ne doit pas être fragmenté. Il peut être utilisé par exemple, par des terminaux sans mémoire de masse qui chargent leur OS à l'aide du protocole TFTP. En effet le logiciel contenu en ROM est incapable d'assurer le réassemblage des datagrammes. Le bit MF (More Frags) est mis à 1 tant que tous les fragments d'un même datagramme ne sont pas arrivés. Le dernier fragment à donc ce bit MF à 0. Dans le cas de datagrammes non fragmentés, ce bit est évidemment à 0. Les autres bits de ce champ indiquent la position des données par rapport à leur position dans le datagramme d'origine.
9	TTL	Time To Live : Ce champ représente en secondes la durée de vie d'un datagramme IP. La valeur de départ est de 60. A chaque passage dans un routeur la valeur est décrémentée d'une seconde (pour simplifier le travail). Lorsque la valeur atteint 0, le routeur qui reçoit le paquet le détruit et envoie un paquet ICMP sur le réseau. Ce mécanisme a pour but d'éviter que des datagrammes dont l'adresse est erronée tournent sans fin dans l'internet.
10	Protocol	On trouve dans ce champ le code du protocole utilisé au-dessus de IP. Si la valeur est 6, le protocole qui utilise IP est TCP. Si la valeur est 17, il s'agit d'UDP. Si la valeur est 1, d'ICMP.
11-12	Header Checksum	Ce champ contient une checksum sur 16 bits des octets de l'en-tête IP.
13-16	Source Address	Ce champ contient l'adresse du nœud qui a émis le datagramme IP.
17-20	Dest. Address	Ce champ contient l'adresse du nœud de destination
20 + multiple de 32 bits	Options	Ce champ de longueur variable, mais toujours multiple de 32 bits est utilisé parfois pour définir des informations concernant la sécurité ou le routage

Figure XII-5 : Signification des champs de l'en-tête IP.

Exemple

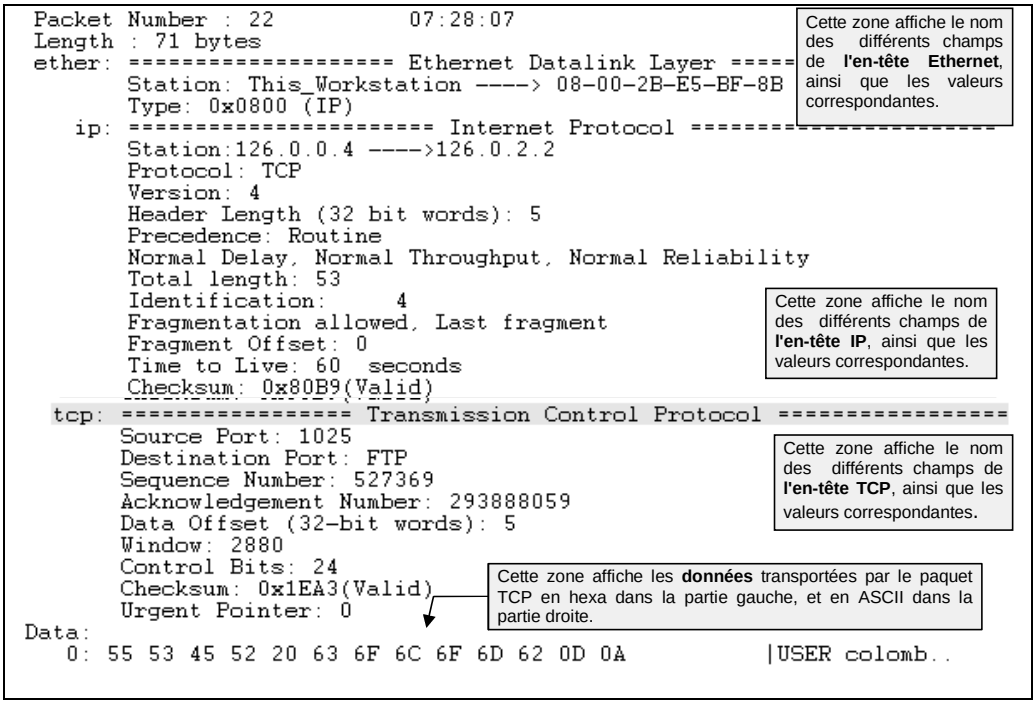


Figure XII-6 : Trame Ethernet avec Datagramme IP affichée sur un analyseur de protocole.

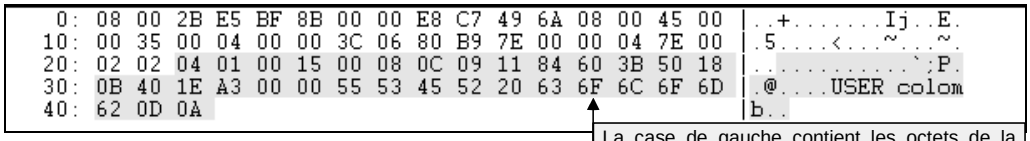


Figure XII-7 : La même trame non décodée, affichée en hexadécimal.

EXERCICE :

Dans l'en-tête Ethernet, quelles sont les 2 adresses MAC en hexa? : ..... , ...

Dans l'en-tête **Ethernet**, quel est le code du protocole et quel est ce protocole : ..... , ...

La trame capturée et analysée ici, est-elle du type Ethernet II ou du type 802.2 ? : .....

Dans l'en-tête **IP**, quelles sont les adresses IP source et destination en notation décimale point et en hexa?

Source en décimal ..... Source en Hexadécimal .....

Destination en décimal ..... Destination en hexadécimal .....

Quel est le protocole indiqué dans cet en-tête ? : .....

Quelle est la longueur de l'en-tête en mots de 32 bits et le nombre d'octets correspondants : ..... , ...

Quel est le numéro de ce datagramme IP ? : .....

L'en-tête IP possède-t-il un champ 'Options' ? : .....

Vos notes:

_____

_____

_____

_____

Mise en forme : Puces et numéros

~~VII -~~XIII - Les protocoles de transport TCP et UDP

Le tableau ci-dessous permet de comparer les 2 protocoles de transport. On utilise l'un ou l'autre de ces protocoles en fonction du niveau de fiabilité réclamée par l'application dont les données sont véhiculées par le protocole de transport.

Caractéristiques	UDP	TCP
Longueur de l'en-tête	8 octets	20 octets
Etablissement d'une connexion	Non	Oui
Séquencement / Perte	Non	Oui
Accusé de réception	Non	Oui
Contrôle de flux	Non	Oui
Multiplexage	Non	Oui
Contrôle d'erreur	Non (Oui en option)	Oui

XIII-A- UDP

XIII- A- 1- Généralités

UDP (RFC 768 de 1980) permet des échanges de paquets de données individuels. Chaque datagramme UDP est encapsulé dans un datagramme **IP**. Celui-ci sera positionné dans une trame conforme au réseau physique utilisé.
Contrairement à TCP, **UDP**, comme IP, est un protocole sans connexion et non fiable. UDP n'assure pas les accusés de réception. Il ne gère pas le séquencement des messages, ni le contrôle de flux. Il peut donc y avoir perte des données ou duplication ou deséquencement. Les fonctions de contrôle doivent donc être assurées par les applications utilisant UDP si nécessaire.
UDP est efficace pour des applications qui utilisent la diffusion (pas de temps de connexion)
Il utilise comme TCP un système d'identification des applications, basé sur des numéros de ports.
Les applications qui utilisent UDP sont :

- TFTP Port 69
- DNS Port 53
- SNMP Ports 161-162
- RIP Port 0208

XIII- A- 2- En-tête

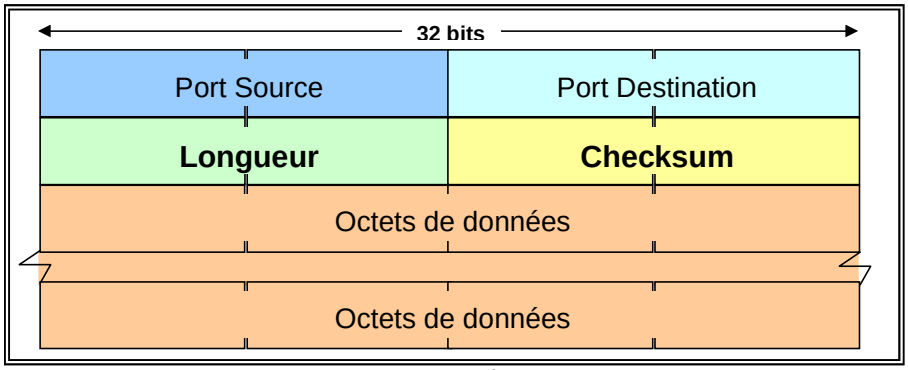


Figure XIII-1 : En-tête UDP.

Octet	Champ	Rôle
1 et 2	Port Source	Numéro de l'application qui émet les données qui sont transportées par le message UDP.
3et 4	Port Destination	Numéro de l'application à laquelle les données sont destinées
5 et 6	Longueur	Longueur du message UDP en octets.
7 et 8	Checksum	Le champ Checksum est utilisé de façon facultative. Sa valeur est souvent à 0000. Pour qu'il soit réellement fonctionnel, il faut que l'application qui utilise UDP en fasse la demande expresse.

Figure XIII-2 : Signification des champs de l'en-tête UDP.

Exercice

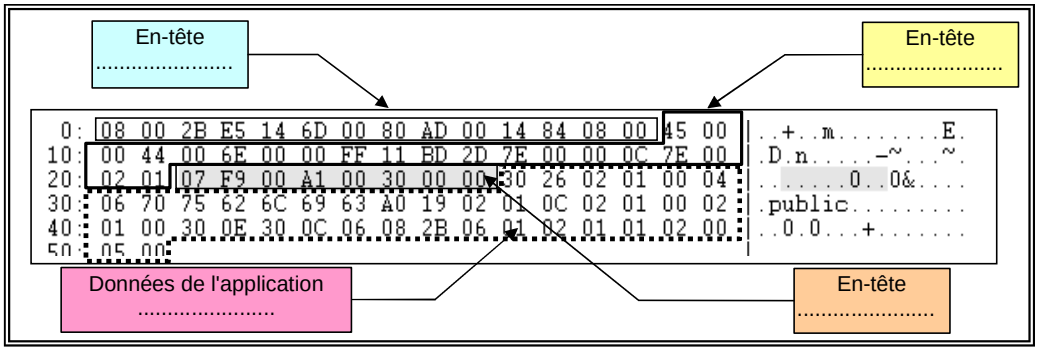


Figure XIII-3 : Analyse de la même trame, mais non décodée.

EXERCICE : Complétez les bulles. En vous aidant de la page précédente, indiquez quelle est la valeur du port correspondant à l'application SNMP destination dans l'en-tête UDP?. : .....

```
Packet Number : 158          07:46:24
Length : 86 bytes
ether: ==================== Ethernet Datalink Layer ====================
       Station: 00-80-AD-00-14-84 ----> 08-00-2B-E5-14-6D
       Type: 0x0800 (IP)
   ip: ======================= Internet Protocol ========================
       Station:126.0.0.12 ---->126.0.2.1
       Protocol: UDP
       Version: 4
       Header Length (32 bit words): 5
       Precedence: Routine
       Normal Delay, Normal Throughput, Normal Reliability
       Total length: 68
       Identification:   110
       Fragmentation allowed, Last fragment
       Fragment Offset: 0
       Time to Live: 255 seconds
       Checksum: 0xBD2D(Valid)
   udp: ===================== User Datagram Protocol =
       Source Port: 2041
       Destination Port: SNMP
       Length = 48
       Checksum: 0x0000(checksum not used)
  snmp: ================ Simple Network Management Protocol ==============
       Message: 38 bytes
       Version: Version-1 (0)
       Community: public
       GetRequest-PDU: 25 bytes
       Request Id: 12
       Error Status: noError
       Error index: 0
       Variable bindings list: 14 bytes
       Variable binding: 12 bytes
       Name:
             iso(1).org(3).dod(6).internet(1).mgmt(2).mib-2(1)
             .system(1).sysObjectID(2).0
       Type: NULL
       Value:NULL
```

Décodage de l'en-tête Ethernet

Décodage de l'en-tête **IP**

Décodage de l'en-tête **UDP**
Port Source (2 octets)
Port Destination (2 octets)
Longueur (2 octets)
Checksum (2 octets)

Champs concernants l'application SNMP

Figure XIII-4 : Analyse et décodage d'une trame contenant un datagramme UDP.

XIII-B- •TCP

XIII- B- 1- Généralités

Pour la transmission de suites de données très longues comme des fichiers par exemple, on met en œuvre un protocole plus fiable nommé **TCP** (RFC 793 de 1981) qui utilise des **circuits virtuels**. Ce sont des connexions full-duplex entre les 2 applications dialoguant sur les 2 ordinateurs. Avant de commencer une transaction, le protocole à chaque extrémité du circuit virtuel ouvre une connexion. Chaque connexion correspond à un port qui comporte un numéro alloué de manière dynamique côté client et de manière bien définie côté serveur. Les segments **TCP** comportent un en-tête et un champ de données. L'ensemble est passé à la couche **IP** qui y ajoute l'en-tête IP, puis le tout est inséré dans une trame.

XIII- B- 2- Format de l'en-tête

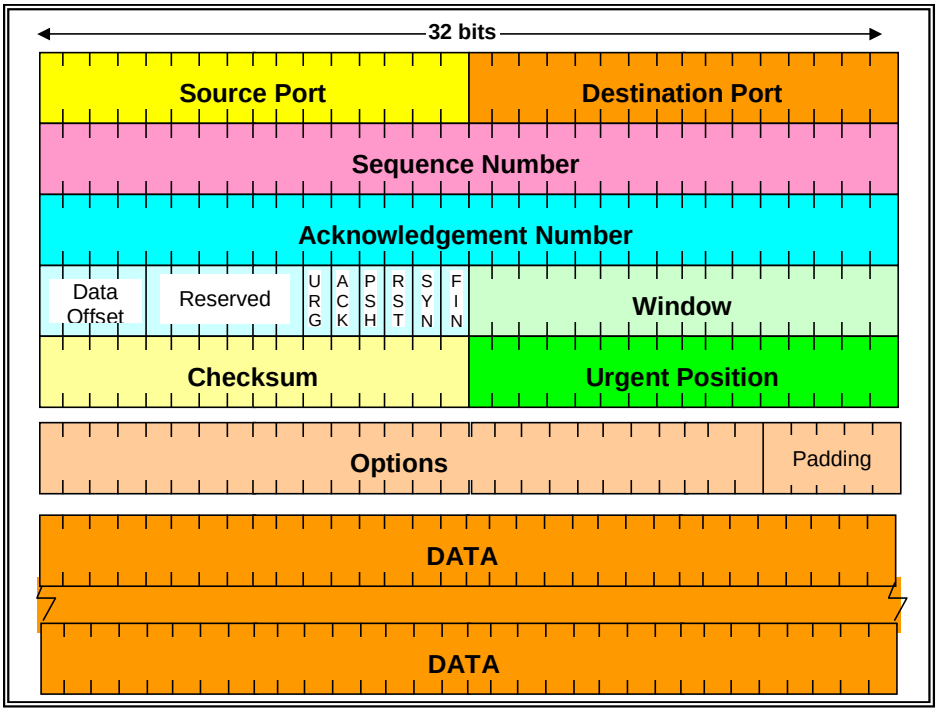


Figure XIII-5 : Format messages TCP.

Octets	Champ	Rôle
1 et 2	Source port	Numéro de l'application qui émet le paquet TCP.
3 et 4	Destination Port	Numéro de l'application à laquelle le paquet est destiné.
5 à 8	Sequence Number	Numéro du premier octet de données transporté dans le paquet.
9 à 12	Ack. Number	Numéro du prochain octet attendu en provenance de la couche TCP distante. Ce champ est équivalent à un accusé de réception dans un paquet transportant des données.
13 et 14	Data Offset et Flags	Ce champ contient la longueur de l'en-tête TCP exprimé en mots de 32 bits. Cette information est nécessaire car il peut exister un champ "options" supplémentaire. **Flags** **ACK** : Ce bit à 1 indique que le paquet TCP est un accusé de réception. **SYN** : Ce bit à 1 est utilisé lors de l'établissement d'une connexion. **FIN** : Ce bit à 1 est utilisé au moment de la fermeture de la connexion. **RST** : Ce bit à 1 indique qu'il y a réinitialisation de la connexion suite à erreurs irrécupérables. **PSH** : Ce bit à 1 impose la remise immédiate des données à la couche supérieure. **URG** : Ce bit à 1 indique que des données urgentes sont placées dans le paquet. Dans ce cas, le champ "Urgent Position" est valide et contient un pointeur qui permet de déterminer la longueur de ces données. Ce type de

		flag est utilisé par exemple au cours d'une session Telnet pour envoyer une commande de contrôle au serveur Telnet
15 et 16	Window	Ce champ contient le nombre d'octets que la session TCP peut encore recevoir sans accusé de réception. Lorsque la valeur atteint 0, l'émetteur ne doit plus émettre. Ce champ permet le contrôle de flux.
17 et 18	Checksum	Checksum du paquet TCP y compris l'en-tête.
19 et 20	Urgent position	Ce pointeur de données urgentes est valide si le flag URG=1.

Figure XIII-6 : Champs de l'en-tête TCP.

XIII- B- 3- Fonctionnalités TCP

- **Accusé de réception**

La fiabilité de la transmission est assurée par l'utilisation d'accusé de réception. Le flag ACK est alors positionné à 1. Le champ Achnowledgement Number est aussi utilisé pour les accusés de réception. Il permet de connaître le numéro du prochain octet attendu par une session TCP et par conséquent le numéro du dernier octet reçu.

- **Séquencement et Détection de pertes de données**

Les champs Sequence Number et Acknowledgement Number permettent aussi de s'assurer que les octets arrivent dans l'ordre correct. Ils permettent aussi de détecter si des octets ont été perdus.

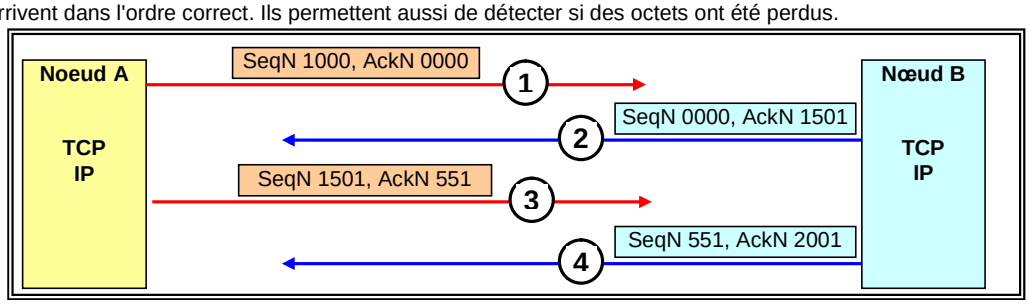


Figure XIII-7 : Utilisation des champs Sequence Number et Ack Number

1- La couche TCP du nœud A émet un paquet avec un Sequence Number à 1000, ce qui veut dire que le premier octet de ce paquet est le numéro 1000. La valeur de Ack Number est de 0000, ce qui veut dire que la couche TCP de A attend le premier octet en provenance de la couche TCP de B.
2- La couche TCP de B émet un paquet avec Seq Number à 0000, ce qui veut dire que le numéro du premier octet de données est 0000. La valeur de Ack Number est 1501, ce qui veut dire que la couche TCP de B a bien reçu les 1500 premiers octets en provenance de A et attend le numéro 1501 et suivants.
3- La couche TCP de A, envoie les octets à partir du numéro 1501. La valeur de Ack Number est 551, ce qui veut dire que A a bien reçu les octets jusqu'à 550 en provenance de B et attend le numéro 551 et suivants.
4- La couche TCP de B envoie les octets à partir de 551. Elle attend l'octet 2001, ce qui veut dire qu'elle a bien reçue les octets jusqu'à 2000 inclus.
Ce mécanisme permet d'accuser réception des octets reçus et de vérifier que l'ordre est respecté. Ainsi B attend l'octet 1501 et qu'il reçoit 2001, cela veut dire qu'il y a deséquencement ou perte d'octets.

- **Contrôle de flux**

Le champ Windows permet d'indiquer à l'émetteur combien d'octets il peut encore envoyer sans accusé de réception. Lorsque la valeur de la fenêtre atteint 0, l'émetteur doit interrompre l'envoi des données. Il peut recommencer le transfert lorsque la valeur de la fenêtre est différente de 0. Ce champ permet donc le contrôle de flux.

- **Multiplexage**

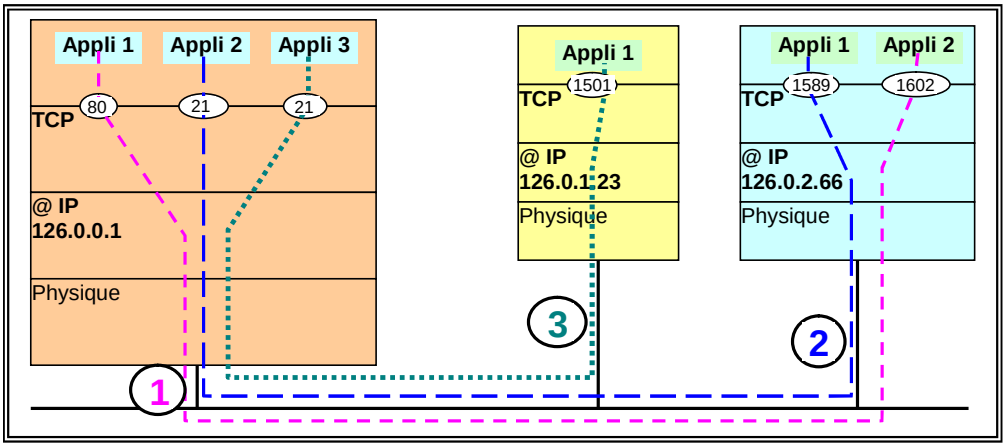


Figure XIII-8 : Multiplexage TCP.

TCP peut servir plusieurs applications dans un même nœud. C'est le multiplexage TCP. Chaque extrémité de la liaison se caractérise par une adresse IP et un numéro de port.
Par exemple l'application 1 dans le nœud dont l'adresse IP est 126.0.0.1 est l'extrémité 126.0.0.1,80 de la connexion avec l'application 2 du nœud 126.0.2.66. Cette autre extrémité est dont notée 126.0.2.66,1602.

- **Connexions**

TCP établit une connexion avant de transmettre des données. Cette opération est relativement longue, car elle nécessite l'envoi de plusieurs paquets de dialogue entre les 2 couches TCP qui veulent établir la connexion.
Chaque connexion est symbolisée par les valeurs des 2 extrémités. Ainsi la connexion **1** est définie par :
(126.0.01,80,126.0.2.66.1602)

Vos notes personnelles :

XIII- B- 4- Exemple TCP

```
Packet Number : 22          07:28:07
Length : 71 bytes
ether: ==================== Ethernet Datalink Layer ====================
       Station: This_Workstation ----> 08-00-2B-E5-BF-8B
       Type: 0x0800 (IP)
   ip: ====================== Internet Protocol ======================
       Station:126.0.0.4 ---->126.0.2.2
       Protocol: TCP
       Version: 4
       Header Length (32 bit words): 5
       Precedence: Routine
       Normal Delay, Normal Throughput, Normal Reliability
       Total length: 53
       Identification:     4
       Fragmentation allowed, Last fragment
       Fragment Offset: 0
       Time to Live: 60  seconds
       Checksum: 0x80B9(Valid)
  tcp: ================= Transmission Control Protocol =================
       Source Port: 1025
       Destination Port: FTP
       Sequence Number: 527369
       Acknowledgement Number: 293888059
       Data Offset (32-bit words): 5
       Window: 2880
       Control Bits: 24
       Checksum: 0x1EA3(Valid)
       Urgent Pointer: 0
Data:
   0: 55 53 45 52 20 63 6F 6C 6F 6D 62 0D 0A          |USER colomb..
```

Figure XIII-9 : Trame Ethernet avec un paquet TCP.

```
 0: 08 00 2B E5 BF 8B 00 00 E8 C7 49 6A 08 00 45 00 |..+.......Ij..E.
10: 00 35 00 04 00 00 3C 06 80 B9 7E 00 00 04 7E 00 |.5....<...~...~.
20: 02 02 04 01 00 15 00 08 0C 09 11 84 60 3B 50 18 |............`;P.
30: 0B 40 1E A3 00 00 55 53 45 52 20 63 6F 6C 6F 6D |.@....USER colom
40: 62 0D 0A                                        |b..
```

Figure XIII-10 : Même trame non décodée.

EXERCICE :

Quelle est l'adresse MAC de la station source ? : .....

Dans la figure ci-dessus, entourez les octets correspondants aux en-têtes **Ethernet**, **IP**, **TCP** et ceux de l'**application**.

Dans l'en-tête **IP**, quel est le code du protocole TCP? : .....

Dans la zone **TCP**, tracez un trait vertical pour séparer les différents champs.

Quelle est la longueur en nombre de mots de 32 bits, puis en octets du segment TCP? : ....., ...

Pour quelle application, TCP est-il employé ? : .....

En vous aidant de la figure précédente, quels sont les codes des applications FTP client et serveur dans l'en-tête TCP? : .....

XIV - Protocoles de résolution d'adresses IP

XIV-A- ARP (Protocole code 0806)

Une adresse IP est liée à une adresse physique en utilisant le protocole **ARP**[28] (RFC 826 de 1982). Pour trouver l'équivalence entre une adresse physique inconnue et une adresse IP connue, une station diffuse un paquet ARP contenant une adresse **IP** destination ainsi qu'une adresse physique source et une adresse IP source. La station destination renvoie un paquet contenant son adresse physique et son adresse IP. Une table de correspondance entre les adresses physiques et les adresses IP, est mise à jour dans chaque station.

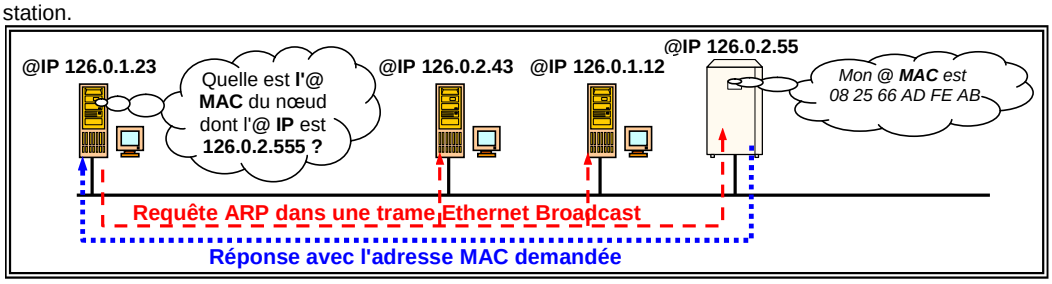


Figure XIV-1 : Requête et réponse ARP.

Supposons qu'un utilisateur sur la station dont l'adresse IP est 126.0.1.23 ouvre une première session Telnet sur le serveur UNIX dont l'adresse IP est 126.0.2.25. La station doit envoyer une trame Ethernet pour ouvrir la session Telnet, mais elle ne connaît pas l'adresse MAC du serveur. Elle va dont envoyer une trame Ethernet Broadcast avec une requête ARP. Cette requête est reçue par tous les nœuds du réseau, mais seul celui dont l'adresse IP correspond à celle indiquée dans la requête va répondre. Dans cette réponse ARP on trouve l'adresse MAC du serveur demandée par la station.
Pour éviter que ce processus ne se renouvelle à chaque ouverture de session avec le serveur, son adresse MAC et mise dans un cache qui reste valide jusqu'à expiration d'un délai fixé par le système d'exploitation.

```
C:\>arp -a
Interface : 125.0.0.1 on Interface 2
  Adresse Internet      Adresse physique      Type
  125.0.0.2             00-10-7b-3c-3b-70     dynamique
```

Figure XIV-2 Contenu d'un cache ARP.

XIV- A- 1- Format d'un paquet ARP

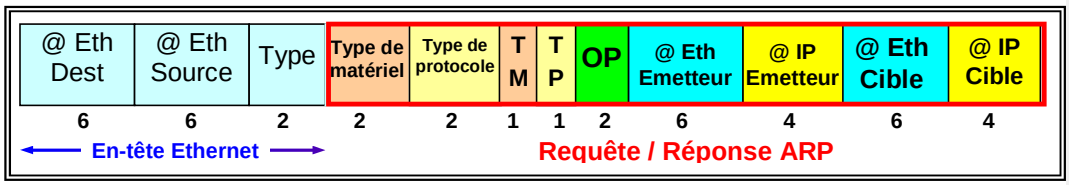


Figure XIV-3 : Format d'un paquet ARP.

Le champ type de l'en-tête Ethernet contient la valeur **0806** qui correspond à ARP.
Dans le paquet ARP qui peut être une Requête ou une Réponse :

- *Type de matériel* : indique le type de trame physique. La valeur est 1 pour Ethernet.
- *Type de protocole* : indique le protocole utilisé dans le système d'adressage. La valeur est 0800 pour IP.
- *TM= Taille matériel* : longueur des adresses MAC utilisées. 6 octets pour Ethernet.
- *TP = Taille protocole* : longueur des adresses utilisées dans le protocole. 4 octets pour IP.
- OP = Opération : est à 1, s'il s'agit d'une requête ARP. Est à 2, s'il s'agit d'une réponse ARP.
- *@ Eth Emetteur* : adresse Ethernet de l'émetteur.
- *@ IP de l'émetteur*.
- *@ Eth Cible* : adresse Ethernet de la cible. Ce champ n'est pas rempli dans la requête, puisqu'il s'agit de l'information recherchée. Par contre, **ce champ est rempli dans la réponse ARP**.
- *@ IP cible*.

[28] ARP = Address Resolution Protocol.

XIV- A- 2- Exemple ARP

```
No.  Source          Destination     Layer  Summary
 15  This_Workstat:  FFFFFFFFFFFF    arp    Req by 126.0.0.4 for 126.0.2.2

Packet Number : 15          07:28:07
Length : 64 bytes
ether: ==================== Ethernet Datalink Layer ====================
       Station: This_Workstation ----> FF-FF-FF-FF-FF-FF
       Type: 0x0806 (ARP)
  arp: ================== Address Resolution Protocol ==================
       Hardware: Ethernet
       Protocol: 0x0800 (IP)
       Operation: ARP Request
       Hardware adress length: 6
       Protocol address length: 4
       Sender Hardware Address: 00-00-E8-C7-49-6A
       Sender Protocol Address: 126.0.0.4
       Target Hardware Address: 00-00-00-00-00-00
       Target Protocol Address: 126.0.2.2

   0: FF FF FF FF FF FF 00 00 E8 C7 49 6A 08 06 00 01 |..........Ij....
  10: 08 00 06 04 00 01 00 00 E8 C7 49 6A 7E 00 00 04 |..........Ij~...
  20: 00 00 00 00 00 00 7E 00 02 02 FF FF FF FF FF FF |......~.........
  30: FF FF FF FF FF FF FF FF FF FF FF FF             |............
```

Figure XIV-4 : Requête ARP.

```
No.  Source          Destination     Layer  Summary
 16  08002BE5BF8B    This_Workstat:  arp    Reply 126.0.2.2=08002BE5BF8B

Length : 64 bytes
ether: ==================== Ethernet Datalink Layer ====================
       Station: 08-00-2B-E5-BF-8B ----> This_Workstation
       Type: 0x0806 (ARP)
  arp: ================== Address Resolution Protocol ==================
       Hardware: Ethernet
       Protocol: 0x0800 (IP)
       Operation: ARP Reply
       Hardware adress length: 6
       Protocol address length: 4
       Sender Hardware Address: 08-00-2B-E5-BF-8B
       Sender Protocol Address: 126.0.2.2
       Target Hardware Address: 00-00-E8-C7-49-6A
       Target Protocol Address: 126.0.0.4

   0: 00 00 E8 C7 49 6A 08 00 2B E5 BF 8B 08 06 00 01 |....Ij..+.......
  10: 08 00 06 04 00 02 08 00 2B E5 BF 8B 7E 00 02 02 |........+...~...
  20: 00 00 E8 C7 49 6A 7E 00 00 04 00 00 00 00 00 00 |....Ij~.........
  30: 00 00 00 00 00 00 00 00 00 00 00 00             |............
```

Figure XIV-5 : Réponse ARP.

EXERCICE : Dans l'en-tête Ethernet, :
- quelle est la particularité de l'adresse de destination MAC ? .....
- notez le code du type de protocole : .....

Dans le paquet ARP requête :
- quel est le code correspondant au type de trame Ethernet ? : .....
- quel est le code correspondant au protocole IP ?: .....
- quelles sont les longueurs indiquées pour les adresses "physiques" et les adresses "protocole"? : .... , ...
- quelle est l'adresse Ethernet cible figurant dans la requête ARP ? : .....
- comparez-la avec celle figurant dans l'en-tête Ethernet . .....
- pourquoi la première trame comporte-t-elle des FF après le paquet ARP requête ? .....
.....

Dans le paquet ARP réponse :
- quelle est l'adresse Ethernet objet de la demande figurant dans la réponse ARP ? : .....

XIV-B- RARP (Protocole code 8035)

XIV- B- 1- Rôle de RARP

Le protocole **RARP**[29] (RFC 903 de 1984), permet au contraire d'**ARP**, à une **station sans disque** ou à un **Terminal X**[30] de connaître son adresse **IP** à partir d'une table de correspondance des adresses physiques et IP maintenue à jour sur un **serveur RARP**. Lorsqu'une machine veut connaître son adresse IP, elle envoie un paquet de "requête" RARP contenant son adresse physique et le serveur RARP lui répond en lui envoyant un paquet RARP réponse contenant l'adresse IP. Pour que le protocole puisse être utilisé, il faut que le réseau comporte au moins un serveur RARP. Celui-ci utilise une table normalement contenue dans le fichier /etc/ethers sous UNIX. Le protocole TFTP est ensuite utilisé pour charger le logiciel dans le terminal à partir du serveur.

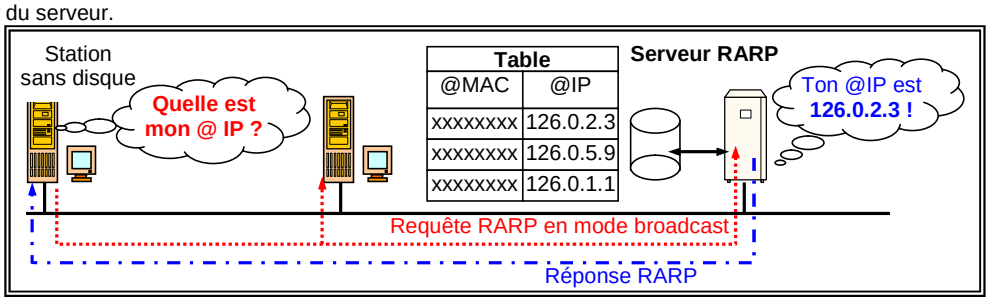


Figure XIV-6 : Requête et réponse RARP.

Supposons que la station sans disque ne puisse pas stocker son adresse IP. Au démarrage, le logiciel contenu dans la ROM de Boot va envoyer une requête RARP dans une trame broadcast. Le serveur RARP consulte la table de correspondance entre les adresses MAC et les adresses IP, établie manuellement par l'administrateur. Il retourne l'adresse IP de la station dans une réponse RARP.

XIV- B- 2- Format RARP

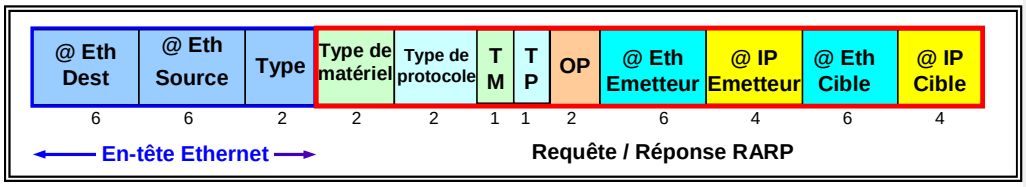


Figure XIV-7 : Format paquet RARP.

Le format d'un paquet RARP est identique à celui d'un paquet ARP. Cependant la valeur du champ type de l'en-tête Ethernet n'est pas O806, mais **8035**.

Dans le paquet RARP qui peut être une Requête ou une Réponse :

- *Type de matériel* indique le type de trame physique. La valeur est 1 pour Ethernet.
- *Type de protocole* indique le protocole utilisé dans le système d'adressage. La valeur est 0800 pour IP.
- *TM= Taille matériel* = longueur des adresses MAC utilisées. 6 octets pour Ethernet.
- *TP = Taille protocole* = longueur des adresses utilisées dans le protocole. 4 octets pour IP.
- OP = Opération ; est à 3, s'il s'agit d'une requête RARP. Est à 4, s'il s'agit d'une réponse RARP.
- *@ Eth émetteur* : adresse Ethernet de l'émetteur.
- *@ IP de l'émetteur* : ce champ est vide dans la requête et contient la valeur donnée par le serveur dans le paquet de réponse.
- *@ Eth cible* : adresse Ethernet de l'émetteur. Ce champ n'est pas rempli dans la requête. Contient l'adresse Ethernet du serveur RARP dans la réponse.
- *@ IP cible* : ce champ est vide dans la requête et contient l'adresse IP du serveur dans la réponse.

RARP ne franchit pas les routeurs. Ce protocole est donc abandonné actuellement au profit de protocoles de configuration IP comme BootP ou DHCP.

[29] RARP = Reverse Address Resolution Protocol = Protocole de résolution d'adresse inverse.
[30] Terminal X = Terminal X Windows connecté à un serveur Unix graphique.

XV - Protocoles de configuration IP automatique

Lorsque les réseaux deviennent très importants, il devient fastidieux de configurer IP sur chaque ordinateur. D'autre part l'usage de plus en plus répandu d'ordinateurs portables interdit sur ces machines des configurations fixes. Il faut qu'au cours des déplacements les ordinateurs portables puissent acquérir une configuration IP (adresse et masque) de manière automatique. L'IETF[31] a développé plusieurs protocoles pour configurer automatiquement les ordinateurs sur les réseaux TCP/IP dont BOOTP[32] et DHCP[33]

XV-A- BOOTP

BOOTP (RFC 951 de 1985 et 1532 de 1993) est un protocole dont le rôle est le même que celui de RARP, mais contrairement à ce dernier, il fonctionne sur les réseaux comportant des routeurs. BOOTP peut être considéré comme une amélioration de RARP et permet d'obtenir des adresses IP pour des stations sans disques ou des terminaux X à partir d'une table, gérée manuellement, existant sur un serveur BOOTP. Le protocole de transport de BOOTP est UDP.

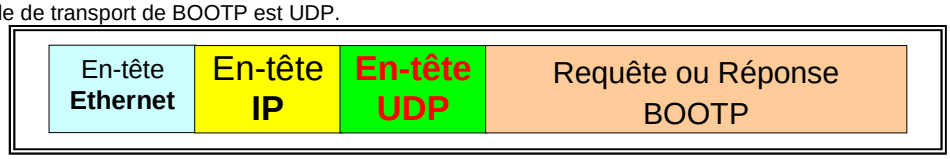


Figure XV-1 : Positionnement de BOOTP dans une trame Ethernet.

Le client BOORP qui désire connaître son adresse IP, envoie une requête dans un paquet UDP.. Le serveur lorsqu'il reçoit la requête consulte la table des adresses et renvoie une réponse BOOTP dans un paquet UDP.
Si le serveur DHCP est sur un autre réseau, un **agent de relais BOOTP** doit être installé sur une station d'un des réseaux.

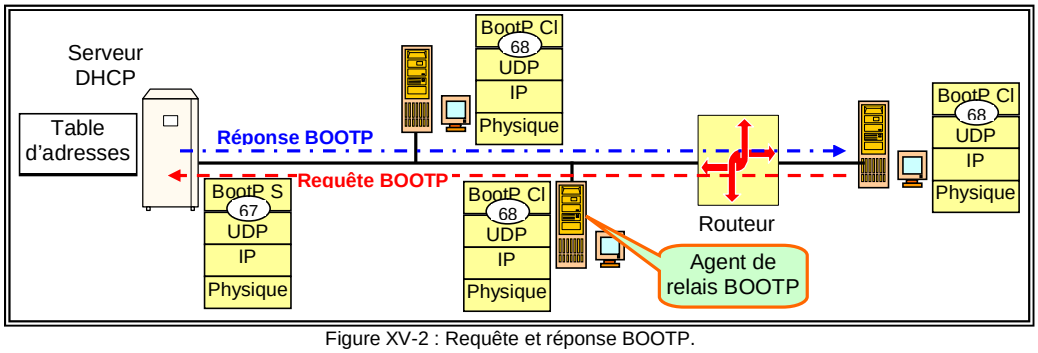


Figure XV-2 : Requête et réponse BOOTP.

En utilisant conjointement BOOTP et le protocole de transfert de fichiers simplifié **TFTP**, on peut obtenir le chargement dans la mémoire de la station ou du terminal du logiciel nécessaire à son fonctionnement. Ce logiciel est stocké dans un répertoire particulier pour chaque station du serveur BOOTP.

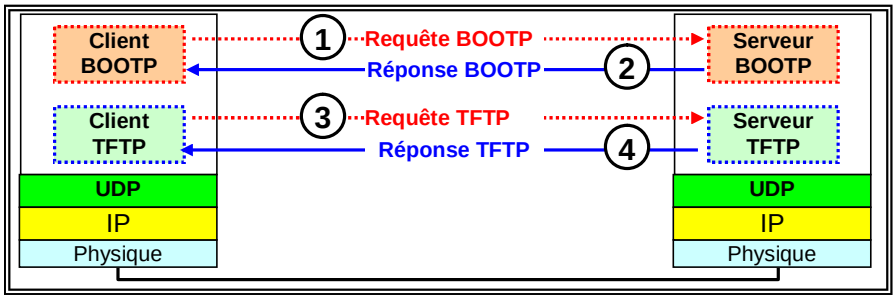


Figure XV-3 : Association de BOOTP et TFTP.

[31] IETF = Internet Engineering Task Force.
[32] BOOTP= Boot Protocol.
[33] DHCP = Dynamic Host Configuration Protocol.

XV-B- DHCP

Le protocole DHCP est une amélioration de BOOTP. Il permet d'allouer des adresses selon plusieurs méthodes et il permet comme BOOTP le chargement de fichiers à partir du serveur. Ce protocole est routable, si on installe un **agent de relais DHCP** dans les routeurs.

Allocation d'adresses

L'allocation des adresses peut se faire de 3 manières différentes :

- **Allocation manuelle** : C'est l'administrateur qui établit, à la main, une correspondance entre l'adresse MAC de l'ordinateur client DHCP et son adresse IP
- **Allocation automatique** : Une adresse est attribuée automatiquement à un ordinateur possédant un client DHCP. L'adresse IP est attribuée au cours de la première requête du client et elle est définitivement attribuée à l'ordinateur.
- **Allocation dynamique** : Cette méthode permet d'attribuer de manière temporaire une adresse IP à un ordinateur qui possède un client DHCP. C'est cette méthode qui est utilisée par les fournisseurs de services Internet pour les clients se connectant par l'intermédiaire du RTC. Cette méthode est aussi très pratique lorsqu'il existe de nombreux ordinateurs portables qui se connectent sur un réseau, puis sur un autre. Un système de gestion de durée des "**Baux**" permet de limiter la durée d'utilisation d'une adresse par un ordinateur.

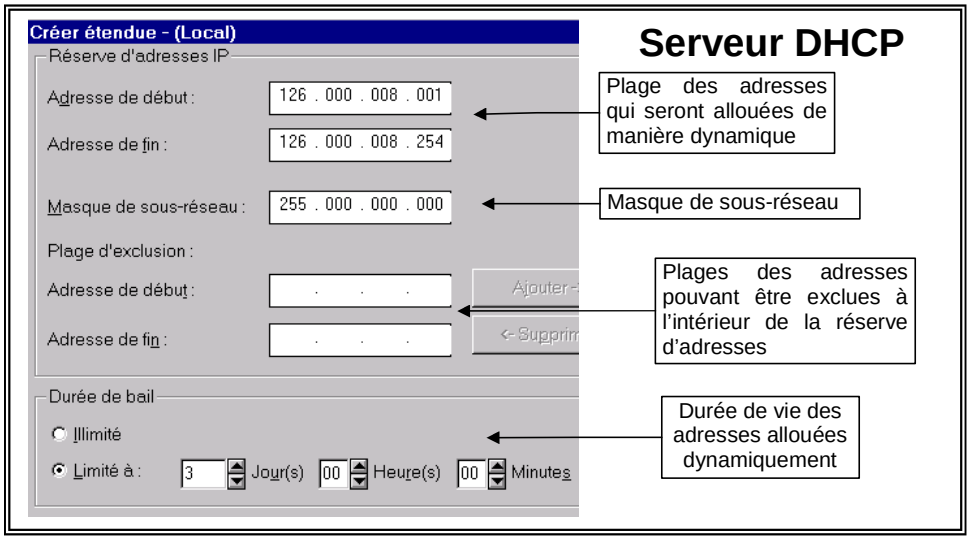


Figure XV-4 : Panneau de configuration d'un serveur DHCP sous NT4.

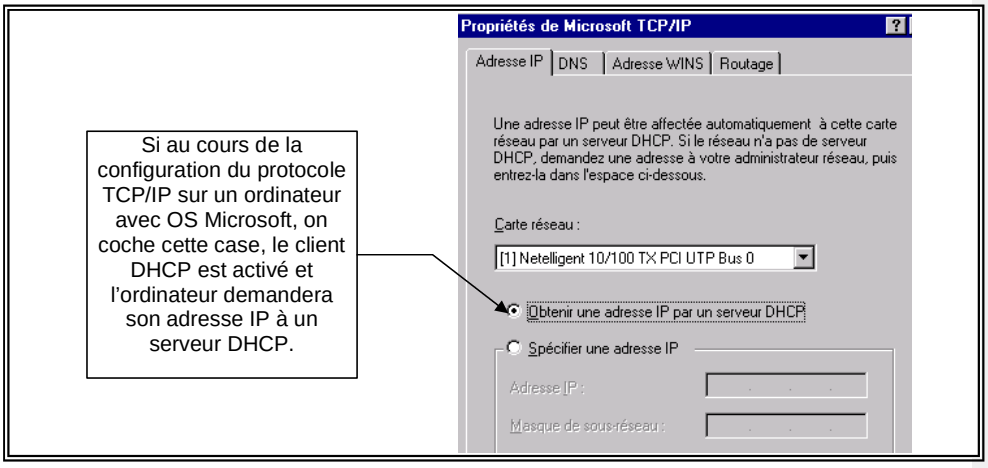


Figure XV-5 : Déclaration du client DHCP dans un ordinateur avec OS Microsoft.

XVI - Routage

Le terme routage désigne la transmission de datagramme à partir d'un nœud d'un réseau vers un autre nœud situé sur le même réseau ou sur un réseau différent. Ce terme désigne aussi le cheminement qui est établi pour transmettre un datagramme IP de son origine vers son point de destination en utilisant les adresses IP contenues dans le datagramme. Un routeur est une machine qui possède au minimum 2 interfaces différentes connectées sur des réseaux différents.

XVI-A- Table de routage

Le fonctionnement du routage nécessite la présence de tables de routage dans les routeurs pour déterminer le chemin à emprunter pour envoyer un datagramme IP d'une machine à une autre à travers l'internet.

La **table de routage** contient les informations suivantes :

- ***l'adresse IP de destination***. Cette adresse peut être une adresse machine ou une adresse réseau suivant la valeur d'un flag H faisant lui-même partie de la table.
- ***l'adresse IP du routeur suivant*** (next hop router) ou l'adresse du réseau qui lui est directement connecté s'il n'y a plus d'autres réseaux.
- des ***flags***. (opion)
- le ***nom de l'interface réseau*** à laquelle il faut envoyer le datagramme pour qu'il soit transmis.

La constitution de cette table est réalisée :

- une seule fois par constitution d'un fichier ou par lancement au démarrage d'une commande particulière (route sous Unix) à l'aide de protocole de routage. On parle alors de **routage statique**.
- avec une remise à jour permanente réalisée par des protocoles spécialisés ***IGP*** comme **RIP**[34], **HELLO**, **OSPF**[35] ou ***EGP*** comme **BGP**[36]. On parle alors de **routage dynamique**.

Les protocoles de type **IGP** (Interior Gateway Protocol) sont utilisés entre routeurs d'un même internet, alors que les protocoles de type **EGP** (Exterior Gateway Protocol) sont utilisés pour des routeurs d'internets différents.

Destination	Gateway	Flags	RefCnt	Use	Interface
125.0.25.36	125.0.1.220	UGH	x	xxxxxx	eth0
125.0.25.75	125.0.1.220	UGH	x	xxxx	eth0
127.0.0.1	127.0.0.1	UH	x	xxxx	lo0
125.0.1.10	125.0.1.12	U	x	xxx	eth0
default	125.0.1.220	UG	x	xxxx	eth0
126.0.0.0	125.0.1.220	UG	x	xxxx	eth0

Figure XVI-1 : Exemple de table de routage de Soleil 125.0.1.12 obtenue avec la commande "netstat".

- la première colonne contient les adresses des machines reconnues par les protocoles de routage. 127.0.0.0 est l'adresse de test de l'ordinateur SOLEIL. L'adresse "default" est l'adresse par défaut du routeur connecté au réseau 125, permettant la connexion aux autres réseaux.
- la seconde colonne comporte les adresses des routeurs à emprunter.
- la troisième colonne comporte des flags. U (Up) indique que la route est en service. G (Gateway) indique que la route passe par un routeur. H indique que l'adresse de la première colonne est une adresse machine complète.
- Refcnt est le compteur de connexions utilisant la route et Use le nombre d'octets transportés
- *Interface* indique le nom de l'interface machine utilisée sur SOLEIL.

XVI-B- Routeur IP

Le **routeur IP** effectue les traitements suivants :

❶ Recherche dans la table de routage de l'adresse de destination complète de la machine. Si elle est trouvée, le datagramme lui est expédiée directement ou à travers un autre routeur. Dans le cas où cette adresse n'est pas trouvée, il passe à la phase suivante.

❷ Recherche de l'adresse du réseau correspondant à l'adresse de destination. Si elle est trouvée, le datagramme est expédié. Sinon, le routeur passe à la phase suivante.

❸ Recherche d'une adresse de routage par défaut qui doit correspondre à l'adresse d'un routeur connecté au réseau. Si aucune adresse n'est trouvée, un message est délivré.

[34] RIP = Routing Information Protocol
[35] OSPF = Open Shortest Path First Protocol
[36] BGP = Border Gateway Protocol

XVI-C- Direct

Le routage direct est la transmission d'un datagramme d'une station à une autre à l'intérieur d'un même réseau.

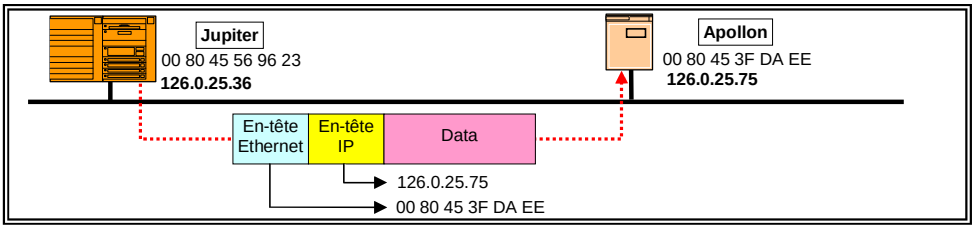


Figure XVI-2 : Routage direct ; Exemple.

Lorsque la couche **IP** reçoit les données à transmettre, elle cherche dans la table de routage si la station de destination est sur le même réseau ou si le datagramme doit transiter par un routeur. Dans l'exemple, la machine destination étant sur le même réseau, le datagramme est passé à la couche liaison qui trouve dans le cache ARP l'adresse physique de destination et l'incorpore dans l'en-tête Ethernet.

XVI-D- Indirect

Le routage indirect fait apparaître la notion de **routeur**. Quand un datagramme est envoyé d'un réseau vers un autre réseau, les parties "réseau" des adresses IP Source (125) et Destination (126) sont différentes. Dans ce cas, la station émettrice envoie le paquet au routeur qui relie les 2 réseaux en utilisant l'adresse physique de ce dernier. Le routeur utilise l'adresse IP pour reconnaître le réseau et la station auxquels il doit envoyer ce paquet. Chaque routeur possède pour chacun des réseaux sur lequel il est connecté, une cache ARP entre les adresses IP et les adresses Physiques.

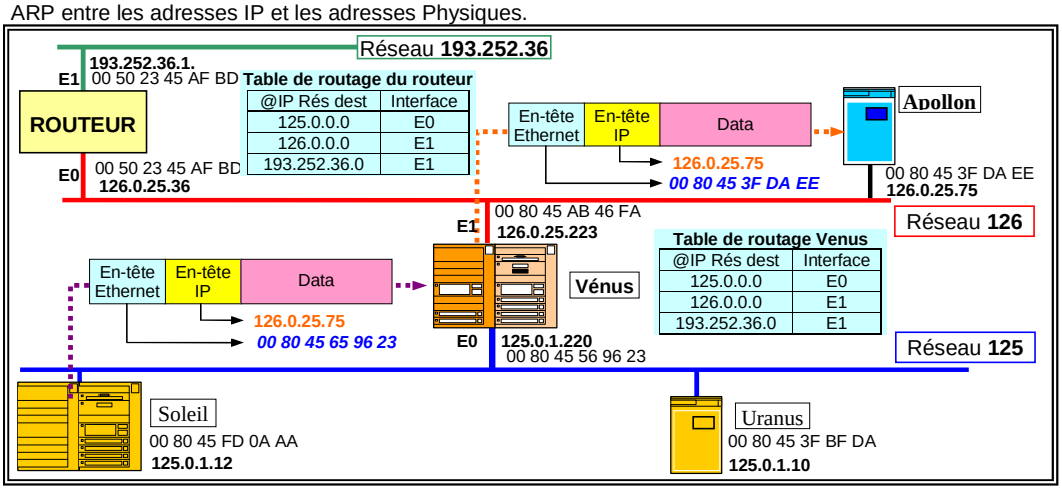


Figure XVI-3 : Routage sur un internet.

Le routeur est la machine **Vénus**. Il comporte 2 cartes réseau, donc 2 adresses MAC et 2 adresses IP comportant des numéros de réseaux différents. Supposons que la machine **Soleil** veuille envoyer un datagramme IP vers la machine **Apollon**. La couche IP recherche l'adresse IP de destination dans la table de routage et déterminer le routeur auquel il faut transmettre le paquet. Le processus suivant est utilisé (comme page précédente) :

-1- Recherche de l'adresse **IP destination complète**, si elle existe dans la table, pour déterminer l'adresse du prochain routeur sur le chemin emprunté pour atteindre la destination.

-2- Si l'adresse complète n'est pas trouvée, la couche IP essaye d'utiliser **l'adresse réseau** destination (126) et le routeur correspondant.

-3- Si l'adresse réseau destination n'est pas non plus trouvée dans la table, IP utilise l'adresse du **routeur par défaut** (125.0.1.220).

En utilisant le contenu du cache ARP présent dans **Soleil**, la couche IP envoie le datagramme dans une trame Ethernet avec comme adresse destination MAC celle du routeur **Vénus** connectée au réseau 125 (00 80 45 56 96 23). L'adresse de destination IP est celle d'**Apollon**. Dans le routeur, IP analyse l'adresse de destination IP. Après consultation des tables de routage et du cache ARP, une trame Ethernet avec l'adresse MAC d'Apollon (00 80 45 3F DA EE) est envoyée sur le réseau 126 avec comme adresse de destination IP 126.0.25.75.

XVII - Protocoles de routage

Pour assurer le routage dynamique et mettre à jour automatiquement les tables de routage, IP utilise 2 grands types de protocoles : les protocoles de passerelles intérieurs (**IGP =** Interior Gateway Protocol) et les protocoles de passerelles extérieurs (**EGP =** Exterior Gateway Protocol). Les IGP sont utilisés à l'intérieur des inter-réseaux ayant une administration commune appelée **systèmes autonomes**. Les protocoles de types EGP sont utilisés entre les systèmes autonomes. Le mot Gateway peut être traduit par passerelle ou routeur.

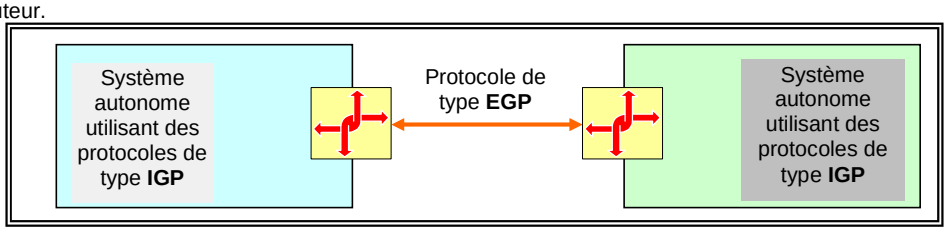


Figure XVII-1 : Protocoles IGP et EGP.

XVII-A- Protocoles de passerelles intérieurs de type IGP

Les protocoles de type IGP sont au nombre de 2 :

- le protocole **RIP** = Routing Information Protocol
- et un protocole plus récent le protocole **OSPF** = Open Shortest Path First Protocol.

XVII- A- 1- RIP

RIP (RFC 1058 de 1988) utilise des requêtes et des réponses pour échanger avec les routeurs voisins des chemins pour compléter ou mettre à jour les tables de routage. C'est un protocole du type "**à vecteurs de distance**", c'est-à-dire qu'il établit les chemins en tenant compte du nom minimum de routeurs à traverser. Un champ dans ce message RIP, appelé **métrique,** indique le nombre de sauts (routeurs) qu'il faut faire pour atteindre la machine de destination. La valeur maximale est de 15. RIP 2 (RFC 1388 de 1993) est une version plus récente et améliorée de RIP.

XVII- A- 2- OSPF

OSPF (RFC 1247 de 1991) est un autre protocole utilisé dans les systèmes autonomes, mais qui n'utilise pas la notion de compteurs de sauts. C'est un protocole du type "**à état de liaison**", c'est-à-dire que les chemins sont établis en tenant compte du **coût** minimum pour les établir. Le coût est calculé en fonction de plusieurs paramètres, dont le débit des lignes utilisées. Ainsi une ligne WAN à un coût supérieur à un réseau LAN. Chaque chemin est entré dans la table de routage avec son coût. Le routeur sélectionne le chemin qui possède le coût le moins élevé.

XVII-B- Protocoles de passerelles extérieurs de type EGP

IP comporte 2 protocoles de passerelle extérieurs : le protocole **EGP** (Exterior Gateway Protocol) et le protocole **BGP** (Border Gateway Protocol), plus récent qui devrait remplacer le premier.

XVIII - Protocole de gestion des erreurs ICMP

ICMP (Internet Control Message Protocol RFC 792 de 1981) est un protocole qui permet l'envoi des messages d'erreur.

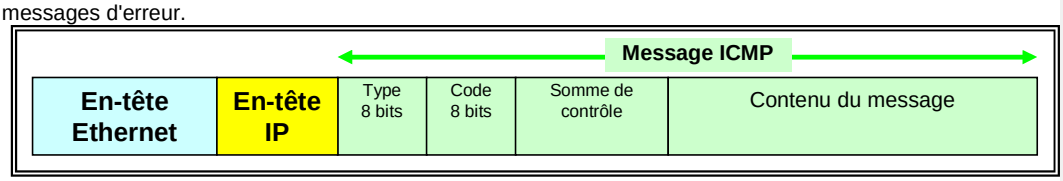


Figure XVIII-1 : Format du message ICMP.

Le champ *type* indique le type de message contenu dans le paquet. Dans chaque type, il peut exister un sous-type indiqué par le contenu du champ *code*. Par exemple, le type 3 indique que la destination est inaccessible. Le code 0 précise qu'il s'agit d'un réseau inaccessible. Les combinaisons type + code indiquent s'il s'agit d'une requête ou d'une réponse.
Le champ *somme de contrôle* est une Checksum.
Les messages ICMP sont utilisés par exemple par l'application Ping pour tester les liaisons. Ils sont générés aussi pour obtenir un masque de sous-réseau, pour obtenir l'heure courante sur une autre machine, pour signaler des ports, des machines ou des réseaux inaccessibles et d'une manière plus générale pour prévenir de toute erreur survenant sur le réseau.

Vos notes :

XIX - Protocoles de lignes séries

Le protocole **IP** est prévu pour fonctionner sur des réseaux LAN. Dans le cas de liaisons séries, 2 protocoles en capsulant les paquets IP sont utilisés : **SLIP** et plus récemment **PPP**.

XIX-A- SLIP

SLIP (Serial Line Internet Protocol RFC 1055 de 1988) permet l'encapsulation de datagrammes IP sur des liaisons séries. Il n'assure aucun adressage, ni séquencement, ni détection ou correction d'erreur. Il permet le transport d'un point à un autre sur une ligne série de paquet IP encadrés de délimiteurs. Il est très utilisé pour les liaisons entre les particuliers et le réseau Internet par RTC et modems.

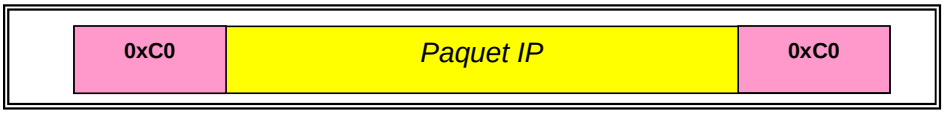


Figure XIX-1 : Format d'une trame SLIP.

SLIP est un protocole orienté caractères, c'est-à-dire que les contrôleurs de communication recherchent des caractères de contrôle (comme 0xC0) dans les données transmises. Si le code C0 du délimiteur existe à l'intérieur du paquet IP, il pourrait être confondu avec le caractère de fin de trame SLIP. Il est donc remplacé par 2 octets de valeur 0xDB et 0xDC. Ces 2 octets sont remplacés par la valeur 0xC0 à la réception de la trame.
Il existe une variante de SLIP qui est CSLIP. Cette autre version compresse une partie de la trame SLIP et permet d'améliorer le débit apparent sur la liaison série.

XIX-B- PPP

Protocole **PPP** (Point To Point Protocol RFC 1331 DE 1992)
Ce protocole plus évolué que SLIP est en passe de le remplacer. Il utilise une variante d'HDLC pour assurer une liaison point à point fiable. Pour assurer la détection et la correction d'erreur, un champ FCS est présent dans la trame et des procédures de retransmission sont prévues.

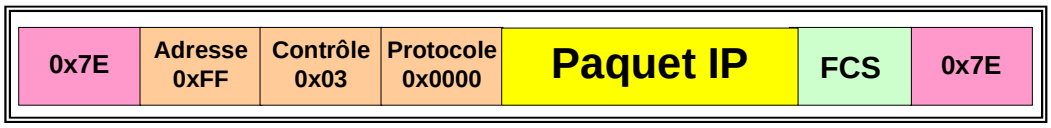


Figure XIX-2 : Format d'une trame PPP.

Des protocoles peuvent être utilisés pour assurer avant transmission des négociations entre les couches des stations d'extrémité. Un ensemble de protocoles est utilisé à cet usage et porte le nom de NCP (Network Control Protocol). Le code du protocole utilisé à un moment de la négociation figure dans le champ "Protocole" de la trame PPP. Les en-têtes PPP peuvent être compressés ainsi que les en-têtes TCP/IP (Méthode Van Jacobson) en utilisant le protocole de négociation IPCP (Internet Protocol Control Protocol). Les champs "Adresse" et "Contrôle" étant inutiles la liaison établie, le format de la trame après compression devient :

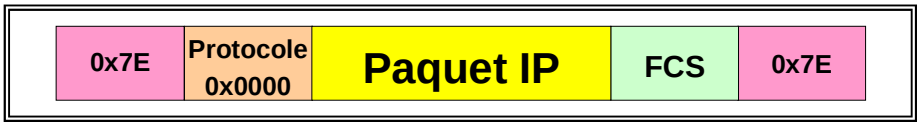


Figure XIX-3 : Format d'une trame PPP avec compression des en-têtes.

Le protocole PPP est très utilisé par les logiciels permettant la connexion à des serveurs **Internet** au moyen de liaisons séries composées de modems et de lignes téléphoniques commutées.

XX - Outils de maintenance

Pour assurer la maintenance des réseaux et matériels fonctionnant sous TCP/IP, il existe un certain nombre d'outils logiciels. Ces outils existent pour tous les systèmes d'exploitation UNIX, NetWare et Microsoft. La syntaxe utilisée ci-après est celui de l'outil Microsoft.

XX-A- Ping

C'est l'outil de maintenance IP de base. Il envoie un message ICMP dans un paquet IP. Ce message doit être retourné par le nœud dont l'adresse est indiquée derrière la commande. Des paramètres permettent de modifier la taille des paquets envoyés, leur nombre, le délai les séparant. Si la trame envoyée n'est pas retournée dans un temps imparti, il y a **time out** et affichage d'un message d'erreur.

- En tapant Ping 127.0.0.1, la carte réseau n'est pas testée, les différentes couches de la pile IP le sont.
- En tapant Ping, suivi de l'adresse IP de l'ordinateur sur lequel est exécuté le test, on teste la pile IP et la carte réseau qui est en mode bouclage.
- En tapant Ping, suivi d'une adresse IP d'un nœud du réseau existant et fonctionnel, on doit obtenir des messages qui affichent le temps mis pour exécuter l'aller-retour entre l'ordinateur local et le distant. Un pourcentage d'erreur est affiché éventuellement.

```
C:\>ping -n 10 -l 500  126.0.0.1

Pinging 126.0.0.1 avec 500 octets de données :

Réponse de 126.0.0.1 : octets=500 temps<10ms TTL=127
Réponse de 126.0.0.1 : octets=500 temps<10ms TTL=127
Réponse de 126.0.0.1 : octets=500 temps<10ms TTL=127
Réponse de 126.0.0.1 : octets=500 temps<10ms TTL=127
Réponse de 126.0.0.1 : octets=500 temps<10ms TTL=127
Réponse de 126.0.0.1 : octets=500 temps<10ms TTL=127
Réponse de 126.0.0.1 : octets=500 temps<10ms TTL=127
Réponse de 126.0.0.1 : octets=500 temps<10ms TTL=127
Réponse de 126.0.0.1 : octets=500 temps<10ms TTL=127
Réponse de 126.0.0.1 : octets=500 temps<10ms TTL=127
```

Figure XX-1 : Exemple de commande Ping.

XX-B- Tracert

Cet outil est connu sous le nom de Trace route sous Unix. Il permet de suivre le chemin emprunté sur un internet pour atteindre un nœud distant. Le logiciel affiche plusieurs durées qui ont été nécessaires pour que les paquets atteignent un nœud déterminé. La ligne se termine par le nom DNS du nœud.

```
C:\>tracert www.renault.fr

Trace l'itinéraire vers www.renault.fr [194.51.107.102]
avec un maximum de 30 tronçons :

  1   150 ms   141 ms   140 ms  192.168.12.1
  2   130 ms   140 ms   140 ms  AUB5.rain.fr [195.101.8.1]
  3   140 ms   140 ms   141 ms  AUBG2-p0-2.rain.fr [194.250.89.133]
  4   140 ms   141 ms   140 ms  ARC5-POS-6-0-0.rain.fr [194.51.221.78]
  5   140 ms   141 ms   140 ms  Sgip2.rain.fr [195.101.10.22]
  6   141 ms   150 ms   140 ms  styx.sgip.fr [194.206.15.194]
  7   160 ms   150 ms   150 ms  www.renault.fr [194.51.107.102]

Routage terminé.
```

Figure XX-2 : Tracert sur www.renault.fr.

Des logiciels graphiques reprennent cet outil pour tracer une carte du chemin emprunté pour atteindre le nœud cible.

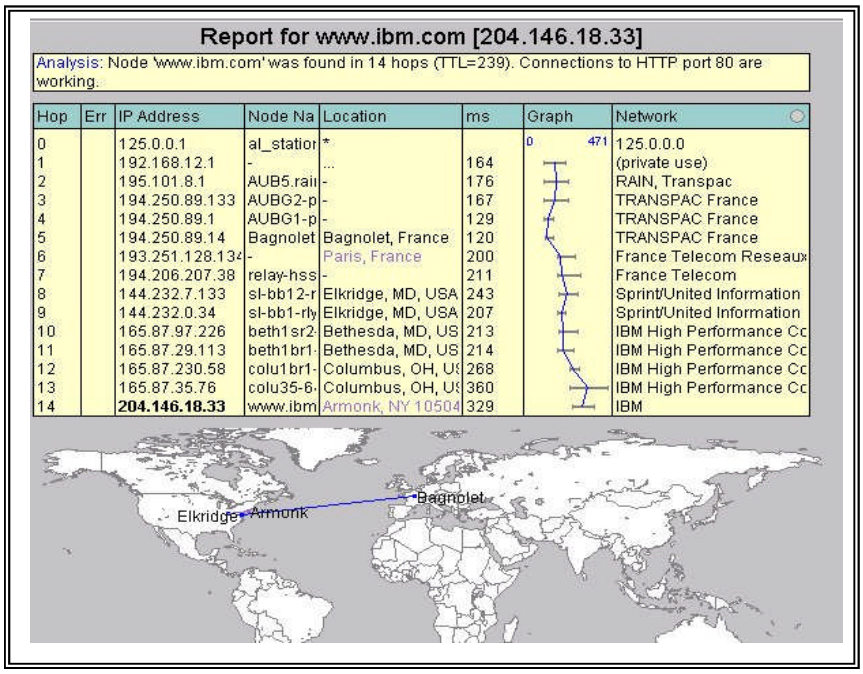


Figure XX-3 : Utilisation de Tracert dans le logiciel graphique Visualroute.

XX-C- Ipconfig

Sos NT cette commande avec l'option **/all** affiche la configuration réseau et TCP/IP de l'ordinateur.

```
C:\>ipconfig /all
Configuration IP de Windows NT
        Nom d'hôte . . . . . . . . . . . . : al_sta
        Serveurs DNS . . . . . . . . . . . :
        Type de nœud. . . . . . . . . . . . : Hybride
        Id d'étendue NetBIOS . . . . . . . :
        Routage IP activé. . . . . . . . . : Oui
        WINS Proxy activé. . . . . . . . . : Non
        Résolution NetBIOS utilisant DNS . : Oui

Ethernet carte El90x1 :
        Description. . . . . . . . . . . . : 3Com 3C90x Ethernet Adapter
        Adresse physique . . . . . . . . . : 00-A0-24-A6-D1-22
        DHCP activé. . . . . . . . . . . . : Non
        Adresse IP . . . . . . . . . . . . : 125.0.0.1
        Masque de sous-réseau. . . . . . . : 255.0.0.0
        Passerelle par défaut. . . . . . . : 125.0.0.2
        Serveur WINS primaire. . . . . . . : 126.0.0.1
```

Figure XX-4 : Résultat de la commande IPconfig sous Windows NT.

Cette commande est remplacée par utilitaire WINIPCFG sous Windows 95.

XX-D- Netstat

Netstat.exe affiche de nombreuses statistiques sur les protocoles et les connexions réseau. Cette commande possède de nombreux paramètres qui donnent des résultats très différents.

```
C:\>netstat –e
Statistiques interfaces
                           Reçu            Envoyé
Octets                    13542338        544425
Paquets unicast              9774          6648
Paquets non-unicast            70            70
Rejets                          0             0
Erreurs                         0             0
Protocoles inconnus             0

C:\>netstat -n
Connexions actives
  Proto  Adresse locale         Adresse extérieure     Etat
  TCP    125.0.0.1:139          126.0.0.1:1042         ETABLIE
  TCP    125.0.0.1:1025         126.0.0.1:139          ETABLIE
  TCP    125.0.0.1:1031         126.0.0.1:139          ETABLIE
  TCP    127.0.0.1:1026         127.0.0.1:1030         ETABLIE
  TCP    127.0.0.1:1030         127.0.0.1:1026         ETABLIE
```

Figure XX-5 : Résultats de la commande Netstat avec 2 options différentes

XX-E- Arp

La commande **arp -a** permet d'obtenir le contenu du cache ARP.

```
C:\>arp -a

Interface : 125.0.0.1 on Interface 2
  Adresse Internet      Adresse physique      Type
  125.0.0.2             00-10-7b-3c-3b-70     dynamique
```

Figure XX-6 : Résultats de la commande arp -a.

Bibiographie

Nom de l'ouvrage	Auteur	Editeur	Commentaires
TCP/IP Le Macmillan 199x	Karnjit S. Siyan	S&SM ISBN : 2-7440-0391-3 Web : http://www.ssm.fr	Très bon ouvrage sur la théorie et les concepts de TCP/IP. Rappels sur les réseaux. Une référence. Questions de révisions à chaque chapitre.
TCP/IP Règles et protocoles	W. R. Kernighan	Addison- Wesley France ISMN 2-87908-82-7	Bon ouvrage plus ancien que le précédent. Questions de révision à chaque chapitre.
TCP/IP Préparation au MCSE Examen 70-059	Rob Scrimger & Kelli Adam	Campus Press ISBN : 2-7440-0598-3 Web : http://www.campuspress.fr	Cet ouvrage est destiné aux personnes voulant se présentant au MCSE. Pas de théorie sur TCP/IP, uniquement la mise en œuvre de TCP/IP sur les produits Microsoft et NT en particulier.
TCP/IP	Craig Zacker	Collection L'expert de Sybex ISBN : 2-7361-3102-9 Web : http://www.sybex.fr	Pas beaucoup de théorie, mais parle de la mise en oeuvre de TCP/IP sur les produits Microsoft et Novell.