Chapitre 8

Accès à Internet et à TCP IP

8.1. Introduction

Un ordinateur personnel permet d'accéder au réseau Internet (cf. paragraphe 1.8.3.). Ce chapitre présente les différents composants et protocoles mis en œuvre pour s'y raccorder.

Il faut en tout premier lieu prendre un abonnement auprès d'une société offrant des accès Internet (un prestataire de service Internet) ou d'un adhérent Internet qui accepte de vous héberger, par exemple le service informatique et réseau de votre établissement. Le raccordement peut se faire par différents moyens (cf. figure 8.1.) en utilisant différentes voies physiques (X25, Numéris, réseau téléphonique...). Comme l'ordinateur personnel n'est pas connecté en permanence, le serveur peut aussi rendre des services supplémentaires, tels l'hébergement de boîte aux lettres de l'utilisateur, le stockage des pages personnelles pour le Web...

Le raccordement nécessite bien sûr une voie physique. Le service du réseau téléphonique est le moyen le plus pratique pour un particulier afin d'accéder à un point d'accès géré par la société. Un modem est nécessaire pour établir la connexion téléphonique puis transmettre et recevoir des données. Le point d'accès n'est autre qu'un routeur du réseau sur lequel on a souscrit un abonnement (droit d'accès). Pour que le transfert de données soit effectif, il faut aussi que soit installé sur l'ordinateur personnel la pile de protocoles TCP/IP utilisée dans Internet.

TCP/IP, protocoles initiaux d'Internet, ont été développés dans les années 1970 à l'Université Stanford et à BBN (Bolt, Beranek et Newman Inc.) sous l'impulsion du DARPA (DoD[1] Advanced Research Project Agency). Ils ont d'abord formé la base d'ARPANET, Advanced Research Project Agency NETwork, pour devenir un standard de facto, au milieu des années 80, de l'interconnexion des réseaux.

1. Le Departement of Defense, DoD, finance de nombreuses activités de recherche aux Etats-Unis.

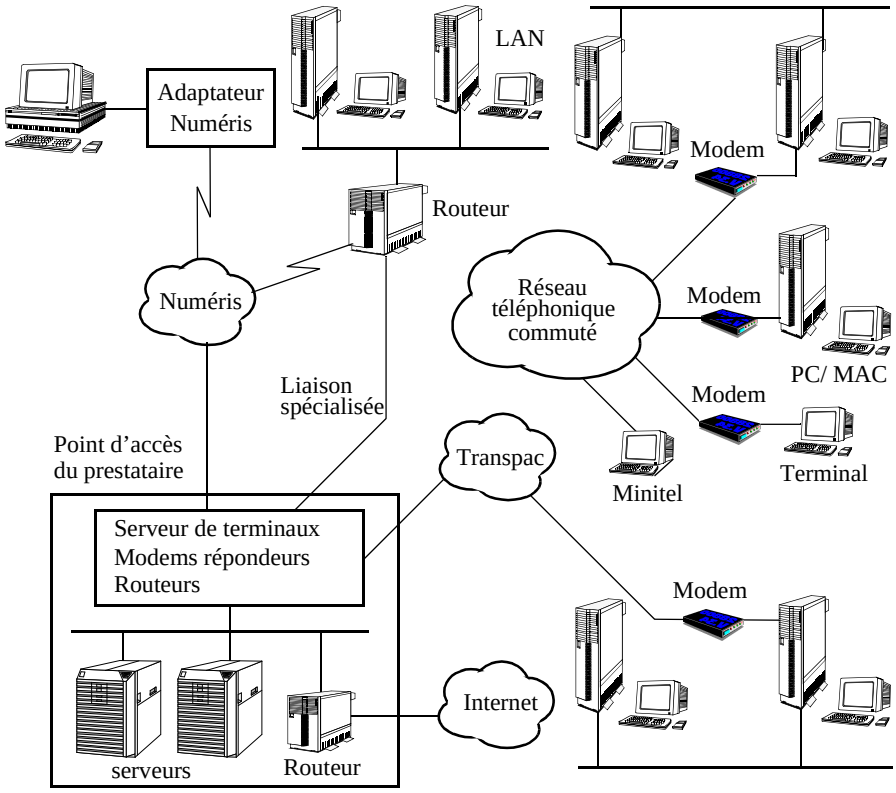

Figure 8.1. *Différents modes de raccordement à Internet*

La couche physique et la couche liaison ne sont pas spécifiées, c'est-à-dire que diverses voies de communication peuvent être utilisées par IP. IP peut donc utiliser une voie virtuelle X25, une liaison téléphonique, une liaison spécialisée, un réseau local... pour son raccordement avec un voisin. Pour les liaisons point à point à faible débit (de type série RS232 ou V24 entre 1200 bits et 19.2 kb/s), typiquement une voie téléphonique avec un modem, un protocole spécifique appelé SLIP (*Serial Link Internet Protocol*) est utilisé. Il y a aussi toute une batterie de protocoles adaptés à des voies physiques telles que : satellites, lignes téléphoniques, radio…

Les protocoles de transport utilisés avec UNIX sont TCP (*Transport Control Protocol*), UDP (*User Datagram Protocol*) et IP (*Internetwork Protocol*). Ils ont été intégrés au sein d'UNIX au début des années 1980. Issus des travaux du projet américain Arpanet, ces protocoles ont été normalisés par le DoD (Department of Defense) dans une série de documents appelés RFC (*Request For Comments*). La figure 8.2. montre les principaux constituants de TCP/IP et les met en relation avec le modèle de référence. La figure 8.3. liste les principaux RFC issus d'Internet. Ces

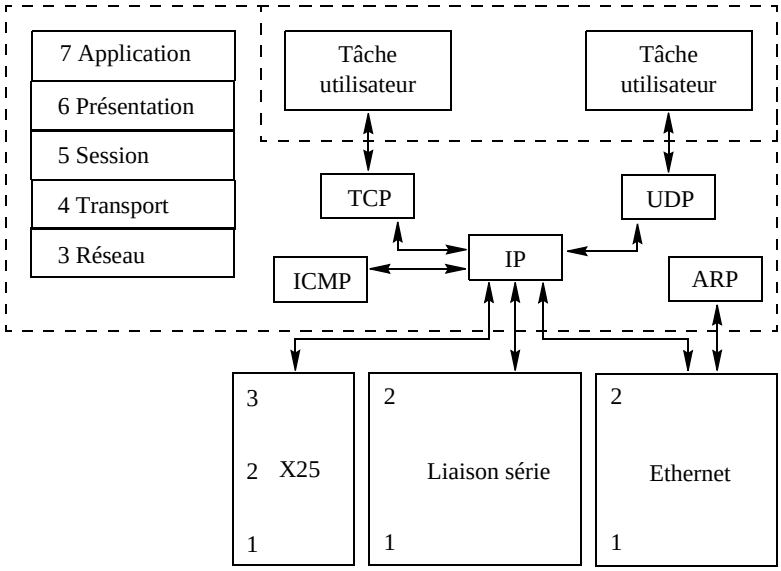

Figure 8.2. *Les protocoles de la famille TCP/IP*

documents sont disponibles auprès du DoD, du NIC (Network Information Center) ou en France auprès de l'INRIA ou dans [10], [26], [35], [42].

Les RFC concernent les standards définissant les protocoles de la pile TCP/IP, des applications telles que le courrier électronique, des informations concernant la mise en œuvre du protocole IP sur d'autres piles protocolaires (Ethernet, Liaison série, réseau X25 (e.g. Transpac), ATM...), l'état de la normalisation, les valeurs que l'on retrouve dans les PDU...

Titre		RFC
Internet Official Protocol Standards	Etat de la standardisation de l'Internet	1720
Assigned Numbers	Valeurs que l'on retrouve dans les PDU.	1700
Host Requirements - Communications	Pile protocolaire nécessaire aux équipements qui peuvent se connecter à l'Internet.	1122
Host Requirements - Applications	Application qui doivent se trouver dans les équipements connecté à l'internet.	1123
Internet Protocol (IP)	Définition du protocole IP	791

Figure 8.3. *Principaux protocoles Internet et RFC correspondantes*

Titre		RFC
Internet Control Message Protocol (ICMP)	Définition du protocole ICMP	792
User Datagram Protocol (UDP)	Définition du protocole UDP	768
Transmission Control Protocol (TCP)	Définition du protocole TCP	793
Telnet Protocol	Définition des messages échangés par les applications telnet (terminal virtuel)	854,855
File Transfer Protocol (FTP)	Définition des messages échangés par les applications ftp (transfert de fichiers)	959
Simple Mail Transfer Protocol (SMTP)	Définition des messages échangés par les applications de courier électronique.	821
Domain Name System (DNS)	Définition des messages échangés par les serveurs de nom (correspondance entre le nom d'un équipement et une adresse IP).	13
Trivial File Transfer Protocol (TFTP)	Définition des messages échangés par les applications ftp (transfert de fichiers avec UDP au moment du *boot*)	1350
Routing Information Protocol (RIP)	Définition du protocole de routage	1058
ISO Transport Service on top of the TCP		1006
Point-to-Point Protocol (PPP)	Définition de la mise en place de IP au dessus d'une liaison série en utilisation le protocole PPP.	1661
PPP in HDLC Framing	Idem	1662
Address Resolution Protocol (ARP)	Définition d'un protocole de résolution d'adresse permettant de mettre en correspondance une adresse IP de niveau 3 et une adresse Ethernet de niveau 2	826
A Reverse Address Resolution Protocol (RARP)	Définition d'un protocole de résolution d'adresse permettant de mettre en correspondance une adresse Ethernet une adresse IP.	903

Figure 8.3. *Principaux protocoles Internet et RFC correspondantes*

Titre		RFC
Internet Protocol on IEEE 802	Définition de la mise en place de IP au dessus d'une liaison série en utilisation le protocole PPP.	1042
Transmission of IP over Serial Lines (SLIP)	Définition de la mise en place de IP au dessus d'une liaison série en utilisation le protocole PPP.	1055
Multiprotocol Interconnect on X25 and ISDN in the Packet Mode.	Définition de la mise en place de IP au dessus d'un réseau X25 comme Transpac.	1356

Figure 8.3. *Principaux protocoles Internet et RFC correspondantes*

— TCP, *Transmission Control Protocol*, offre un service de transport sur connexion. Il garantit le séquencement des données, la récupération des erreurs et assure le contrôle de flux de bout en bout.

— UDP, *User Datagram Protocol*, offre un service transport à datagramme, donc sans séquencement ni correction des erreurs, ni contrôle de flux.

— IP, *Internet Protocol*, est mis en œuvre dans la couche réseau. IP assure un service de routage de datagrammes.

— ICMP, *Internet Control Message Protocol*, sert à gérer les erreurs et la transmission des informations de routages entre les routeurs et entre routeurs et abonnés. Ces messages sont créés par la couche réseau elle-même.

— ARP, *Address Resolution Protocol*, établit la correspondance entre les adresses Internet et les adresses de la couche liaison ou physique (MAC[1], *Medium Access Control*, sur un réseau local). Un abonné envoie un message en diffusion sur le réseau en spécifiant l'adresse Internet. L'abonné correspondant, s'il existe, répond en fournissant son adresse de la couche physique.

— Ethernet : protocole de niveau 1 et 2 utilisé dans les réseaux locaux pour transporter les PDU (trames) sur un bus. Les adresse Ethernet sont sur 6 octets. La taille maximale des trames Ethernet est de 1518 octets.

1. Dans un réseau local ou LAN, Local Area Network, chaque coupleur du réseau local possède en propre une adresse. Cette adresse est unique et universelle mais ne contient aucune information sur la position géographique. Seule cette adresse permet au coupleur du réseau local de recevoir les messages qui lui sont destinés. Cette adresse ne peut être utilisée que sur ce réseau local pour envoyer un message à la machine possédant cette adresse de coupleur. Il s'agit donc d'une adresse de niveau « liaison » qui ne peut être utilisée en dehors du réseau local. Le niveau 3 ne peut donc pas utiliser ce qui n'appartient pas à l'espace d'adresse IP. Il faut donc établir la correspondance entre l'adresse IP et l'adresse de niveau 2 (MAC) pour pouvoir envoyer un message à une station du réseau local. Cette opération est automatique avec ARP.

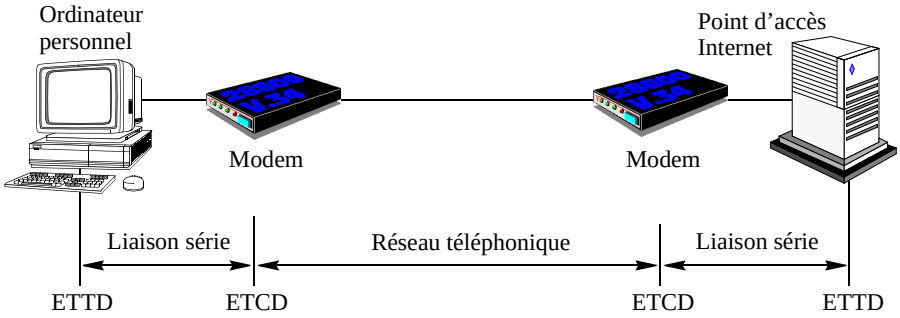

Figure 8.4. *Raccordement au point du réseau Internet*

— X25 : protocole couvrant les niveau 1 à 3 du modèle ISO. Le protocole X25 est utilisé dans les réseaux publics (par exemple Transpac). Il permet de créer un circuit dans le réseau (appelé aussi circuit virtuel) sur lequel des données sont envoyées. Les adresses des réseaux X25 sont sur 14 demi-octets.

— Liaison série : ce support peut être soit un câble série, soit une liaison téléphonique en utilisant un modem.

Ce chapitre est centré sur l'étude des protocoles de niveau 3 et 4, et nous ne nous intéresserons qu'au cas simple de IP sur liaison série en utilisant le protocole SLIP. [10] présente une description détaillée de l'ensemble des protocoles mis en œuvre dans les réseaux Internet. Le paragraphe 8.2. présente les caractéristiques essentielles des liaisons séries (c'est-à-dire le niveau physique dans le modèle de référence OSI). Le paragraphe 8.3. décrit le protocole SLIP permettant d'envoyer des paquets IP sur la liaison série (niveau liaison du modèle de référence OSI). Le paragraphe 8.4. présente le protocole IP et les paragraphes 8.5. et 8.6. décrivent les protocole de niveau 4 TCP et UDP.

8.2. Accès physique

La plupart des ordinateurs personnels (Macintosh, PC...) offrent au moins une prise pour une liaison série afin de connecter une imprimante ou une souris, mais elle représente aussi un moyen économique et parfaitement adapté pour se raccorder à Internet, directement ou par l'intermédiaire du réseau téléphonique et d'un modem (cf. figure 8.4.).

Les protocoles traitant des liaisons point à point sont définis essentiellement par des normes ou des recommandations développés par l'ISO et l'ITU. La terminologie, ETTD et ETCD, est celle que nous avons vue au paragraphe 5.2.2. Seul un sous ensemble des solutions de ces organismes est nécessaire pour le raccordement d'une station à un réseau Internet.

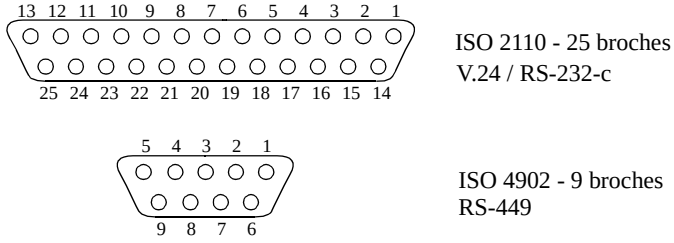

Figure 8.5. *Brochage série normalisé*

8.2.1. *Communication ordinateur modem*

La communication entre un ordinateur (ETTD) et un modem (ETCD) est réalisée grâce à une liaison série. La liaison entre les deux ETCD (modem) se fait, par exemple, en utilisant le réseau téléphonique.

Le raccordement de la liaison série est normalisé. Les connecteurs V.24 (cf. figure 8.5.) possèdent 25 broches car ils sont prévus pour piloter des modems par l'intermédiaire de signaux, en particulier au niveau des horloges d'émission et de réception.

L'utilisateur depuis un menu de configuration de son ordinateur donne les caractéristiques de sa liaison (c'est-à-dire le raccordement entre l'ETTD et l'ETCD) : vitesse de transmission, nombre de bits utiles, parité... Les modems que l'on trouve dans le commerce sont dit intelligents car ils contiennent un processeur qui découvre automatiquement les paramètres de configuration de la ligne en analysant les données qu'il reçoit. Pour éviter un goulet d'étranglement au niveau du modem et une saturation de sa mémoire, qui conduirait à une perte de données, la vitesse de transmission entre ETTD et ETCD devra être supérieure à celle entre les deux ETCD. Le plus simple consiste à utiliser la vitesse maximale offerte par le modem ou l'ordinateur. Ces modems interprètent aussi des ordres simples venant de l'ordinateur.

Pour ces modems, le nombre de broches nécessaire à la communication entre l'ETTD et l'ETCD peut être ainsi réduit. En principe, trois fils dont nécessaires : un fil de masse commune aux deux équipements, un fil pour transmettre les bits de l'ordinateur vers le modem et un autre pour le sens contraire. D'autres fils peuvent simplifier la gestion du modem, comme l'invitation à émettre ou la détection de sonnerie.

Certains fabricants d'équipements informatiques, pour assurer une compatibilité, utilisent toujours la prise V24, mais la tendance va vers l'utilisation d'une prise ayant 9 broches. La figure 8.6. donne le brochage pour les prises 9 broches ainsi que leur position sur le connecteur 25 broches.

La communication avec le modem local étant immédiatement possible, les ordres porteront essentiellement sur la connexion avec le modem distant. Ils vont permettre de définir les caractéristiques de la liaison sur la ligne téléphonique (par exemple :

9 broches	25 broches	Nom RS-232	Sens de transmission	Commentaire
1	8	CD	Vers l'ordinateur	Détection de la porteuse. Le modem a reconnu une porteuse.
2	3	RD	Vers l'ordinateur	Réception des données.
3	2	TD	Vers le modem	Emission de données
4	20	DTR	Vers le modem	L'ordinateur est prêt. Ce signal indique au modem que l'ordinateur est actif.
5	7	M		Masse des signaux
6	6	DSR	Vers l'ordinateur	Le modem est prêt. Ce signal indique à l'ordinateur que le modem est actif.
7	4	RTS	Vers le modem	Demande pour émettre. L'ordinateur indique au modem qu'il veut émettre des données.
8	5	CTS	Vers l'ordinateur	Prêt à émettre. Le modem autorise l'ordinateur à émettre ses données.
9	22	RI	Vers l'ordinateur	Indication d'appel. Le modem reçoit un appel téléphonique (sonnerie).

Figure 8.6. *Affectation des broches des prises séries simplifiées*

imposer une vitesse ou laisser les modems la négocier entre eux, cf. figure 8.8.), demander au modem de composer un numéro de téléphone et attendre une porteuse...

Il existe plusieurs manières de dialoguer avec un modem, la plus répandue utilise le jeu de commandes proposées par le fabriquant de modems Hayes (cf. figure 8.7.). Elles commencent toutes par AT (pour *attention*). La figure 8.7. liste les principales commandes.

Une fois que le modem a pu accéder au réseau téléphonique, composer le numéro de téléphone du modem appeler et que celui-ci a répondu, se synchroniser sur la porteuse de l'équipement distant, nous disposons d'une voie physique bidirectionnelle qui permet d'envoyer des informations sous forme d'octets.

Il faut, pour organiser les échanges, définir au minimum un format de trames. Deux protocoles sont très largement utilisés pour remplir cette mission d'accès à Internet : SLIP (*Serial Line IP*) et PPP (*Point to Point Protocol*). PPP, bien que de plus

Commandes		
ATZ		Initialisation du modem, remise à zéro des paramètres.
ATDTxxxx		Prise de la ligne et numérotation en fréquences vocales du numéro de téléphone `xxxx`.
ATPTxxxx		Prise de la ligne et numérotation par impulsions du numéro de téléphone `xxxx`.
ATBx		Configuration de l'avis de l'UIT :
	Absent x=0	Négociation automatique entre les deux modems de la vitesse ;
	x=3	Mode V.23 (1 200/75 b/s en appel et 75/1 200 b/s en réception) utilisé par le minitel ;
	x=6	Mode V.22 (1 200/1 200 b/s) ;
	x=7	Mode V.21 (300/300 b/s) ;
	x=8	Mode V.22 bis (2 400/2 400 b/s) ;
	x=9	Mode V.32 (9 600/9 600 b/s) ;
	x=10	Mode V.32 bis (14 400/14 400 b/s) ;
	x=11	mode V.32 bis(12 000/12 000 b/s) ;
	x=12	mode V.32 bis(7 200/7 200 b/s) ;
	x=14	mode V.32 (4 800/4 800 b/s).
ATA		Prend la ligne et attend une porteuse.
ATH		Raccroche la ligne. En mode connecté, pour que cette commande soit comprise par le modem local et non pas par l'ordinateur distant, elle doit être précédée de la séquence d'échappement +++
ATQ0 ATQ1		Permet de contrôler les messages de supervision du modem : 0 active l'émission des messages, 1 désactive l'émission des messages : `OK` : la commande a été exécutée correctement par le modem, `CONNECT xxxx` : la connexion a été établie à la vitesse de `xxxx` b/s, `NO CARRIER` : le modem n'a pas pu détecter la porteuse `ERROR` : erreur de syntaxe dans la commande, `BUSY` : le numéro appelé est occupé, `NO ANSWER` : le modem distant n'a pas décroché.

Figure 8.7. *Principales commandes Hayes*

en plus souvent utilisé, ne sera pas étudié du fait de sa complexité. Nous renvoyons le lecteur à [10] pour une étude approfondie de ce protocole.

8.3. SLIP (RFC 1055)

SLIP (*Serial Line IP*) est un moyen très simple d'envoyer des paquets IP sur une liaison série. Le principal problème avec une liaison série est de délimiter les paquets IP. Il n'y a pas de nécessité d'adressage puisque qu'il s'agit d'une liaison point à point.

La méthode d'encapsulation employée par SLIP est rudimentaire. Quand un équipement désire transmettre un paquet, il émet directement son contenu sur la ligne série. Pour indiquer la fin du paquet, il envoie un caractère spécial `END` ayant comme valeur 192 (`0xc0`). Mais :

— si ce code se trouve dans le paquet à émettre pour que le récepteur ne le considère pas comme une fin de paquet, l'émetteur le remplace par la séquence `ESC ESC_END` ayant comme valeur 219, 220 (`0xdb`, `0xdc`) ;

— de même avec le code `ESC`, si ce code se trouve dans le paquet, il est remplacé par la séquence `ESC ESC_ESC` ayant comme valeur 219, 221 (`0xdb`, `0xdd`).

Nous allons étudier à la fin de ce chapitre l'intégration complète de la pile TCP/IP avec SLIP.

8.4. IP

IP, protocole de la couche réseau, offre un service sur datagramme. Chaque datagramme est appelé paquet, dans les RFC, fonctionnant au-dessus de la majorité des protocoles de liaison ou des supports physiques[1] (LAN, *Local Area Network*, LS, Liaison spécialisée...). Il ne garantit pas la livraison des données. Ce service est fourni par les protocoles des couches supérieures, tels que TCP.

IP est le protocole qui construit le service de communication du réseau Internet. C'est un service à datagramme, il ne garantit pas le séquencement des paquets ni ne réalise de correction des erreurs : IP jette les paquets qu'il détecte comme erronés. Le service Internet possède exactement les même propriétés que celles décrites pour le service postal, au paragraphe 2.1.1.

8.4.1. *Structure des paquets*

Comme toute entité protocolaire, IP utilise des PDU. Ces PDU ont une structure précise définie dans le RFC 791. Le format du PDU[2] IP (version 4) est décrit sur la figure 8.9. La description est faite par mots de 4 octets.

---

1. Dans ce paragraphe, on appellera trame le L PDU de la couche liaison. Par exemple, sur le réseau Ethernet la trame, L PDU Ethernet, contient au maximum 1 518 octets dont 18 octets d'en-tête et 1500 octets pour le L SDU.
2. Le terme généralement utilisé dans Internet est paquet, mais on parle aussi de datagramme pour indiquer l'absence de connexion à ce niveau.

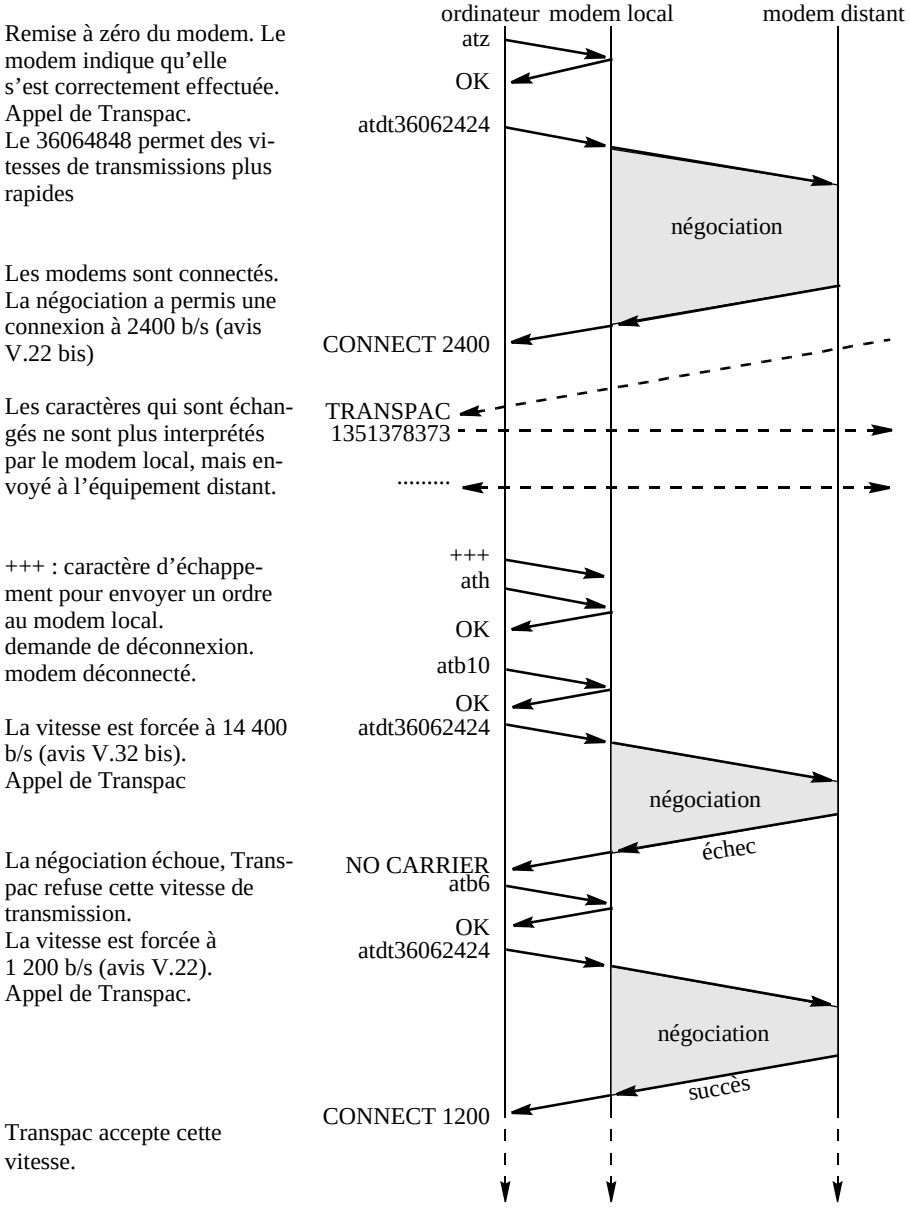

Figure 8.8. *Etablissement d'une connexion téléphonique et négociation des paramètres entre les modems distants*

<table>
<tr><th></th><th colspan="2">1er octet</th><th>2e octet</th><th>3e octet</th><th>4e octet</th></tr>
<tr><td>1er mot</td><td>Version = 4</td><td>Lg en-tête</td><td>Type de service</td><td colspan="2">Longueur du paquet</td></tr>
<tr><td>2e mot</td><td colspan="2">Identification</td><td colspan="3">0 | D | M | place du fragment</td></tr>
<tr><td>3e mot</td><td colspan="2">Durée de vie</td><td>Protocole suivant</td><td colspan="2">Checksum</td></tr>
<tr><td>4e mot</td><td colspan="5">Adresse source</td></tr>
<tr><td>5e mot</td><td colspan="5">Adresse destination</td></tr>
<tr><td>6e mot</td><td colspan="5">options...</td></tr>
<tr><td>...</td><td colspan="5">bourrage</td></tr>
<tr><td>...</td><td colspan="5">SDU - Données utilisateur</td></tr>
</table>

Figure 8.9. *Format du paquet IP en mots de 4 octets (32 bits)*

8.4.1.1. *Version du protocole IP*

Le premier demi-octet indique la version du protocole IP utilisé. Actuellement, la version 4 est utilisée. C'est celle que nous présentons dans ce paragraphe. Une nouvelle version de IP, qui sera la version 6 ou IPng (IP nouvelle génération), est en cours de développement [9].

8.4.1.2. *Longueur de l'en-tête IP*

Le deuxième demi-octet indique la longueur de l'en-tête protocolaire du paquet IP en nombre de mots de 32 bits. Ce champ prend généralement la valeurs 5. L'en-tête fait donc 20 octets et le champ option est vide. Il existe des cas où des options sont nécessaires, par exemple pour forcer les paquets à suivre un chemin imposé par l'émetteur. Dans ce cas, ce champ peut valoir jusqu'à 0 x 0f, soit 15 octets. La partie fixe de l'en-tête restant de 5 mots de 32 bits, le champ option peut faire au maximum 10 mots de 32 bits.

Si les options ne font pas un multiple de 32 bits, des bits de bourrage doivent être prévus pour aligner la partie donnée du paquet.

8.4.1.3. *Type de service IP*

Le champ Type de service, ou ToS (*Type of Service*), est codé sur 8 bits. Ce champ contient des informations qui aideront le routeur à choisir le chemin pour le paquet (cf. figure 8.10.). Par exemple, pour un transfert de gros fichiers, il est préférable de privilégier le débit par rapport au délai d'acheminement. Pour une session interactive, le délai d'acheminement sera primordial.

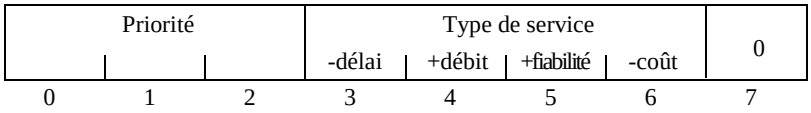

Figure 8.10. *Champ Type de Service*

8.4.1.4. *Taille des paquets IP*

Les paquets IP sont de taille variable, de 0 à 2^{16} octets indiqués dans le champ « longueur du paquet ». En pratique, la taille des paquets est limitée par la taille des SDU supportés par la voie utilisée (trames sur les voies physiques, comme les réseaux locaux, ou liaisons spécialisées, ou voies logiques telle que X25) qui les transportent. La taille de 576 octets est souvent utilisée, pour des raisons historiques, au sein d'Internet.

8.4.1.5. *Identification du paquet IP*

Chaque paquet est identifié par un numéro produit par l'émetteur du paquet. L'émetteur incrémente cette valeur après l'envoi de chaque paquet. Le couple « adresse source - identification » identifie de manière unique un paquet. Cette identification est nécessaire pour détecter les duplications de paquets et la fragmentation.

8.4.1.6. *Fragmentation IP*

La couche réseau doit assurer une indépendance vis-à-vis du support. La taille maximale des PDU de niveau 2 ne doit pas être une limitation à ce niveau. Un mécanisme de fragmentation est disponible pour adapter la taille des paquets à la taille des trames de la voie sur laquelle ils seront émis. On appelle *fragment* la partie d'un paquet adaptée à la taille des PDU de niveau 2.

Le champ « place du fragment » indique l'emplacement du fragment dans le paquet initial. Le mécanisme de fragmentation est mis en œuvre dans tous les routeurs, car l'émetteur ignore le chemin que va suivre son paquet. Si le routeur reçoit un paquet trop grand, il doit l'adapter aux caractéristiques du support qui le transmettra. Pour cela, il effectue une fragmentation des paquets. Il découpe les paquets en fragments de taille plus adaptée qu'il transmet sur le réseau. De l'autre côté, la station destinataire doit réassembler les paquets pour reformer le paquet initial.

La structure du champ contient au début 3 bits, puis la valeur du déplacement dans le fragment initial :

— le premier bit est toujours nul ;

— le deuxième bit DF (*Don't Fragment*), s'il est à 0, indique que le paquet peut être fragmenté par un routeur intermédiaire. S'il vaut 1 et qu'un routeur intermédiaire doit fragmenter, le paquet est rejeté, et un paquet ICMP de contrôle est émis vers le destinataire ;

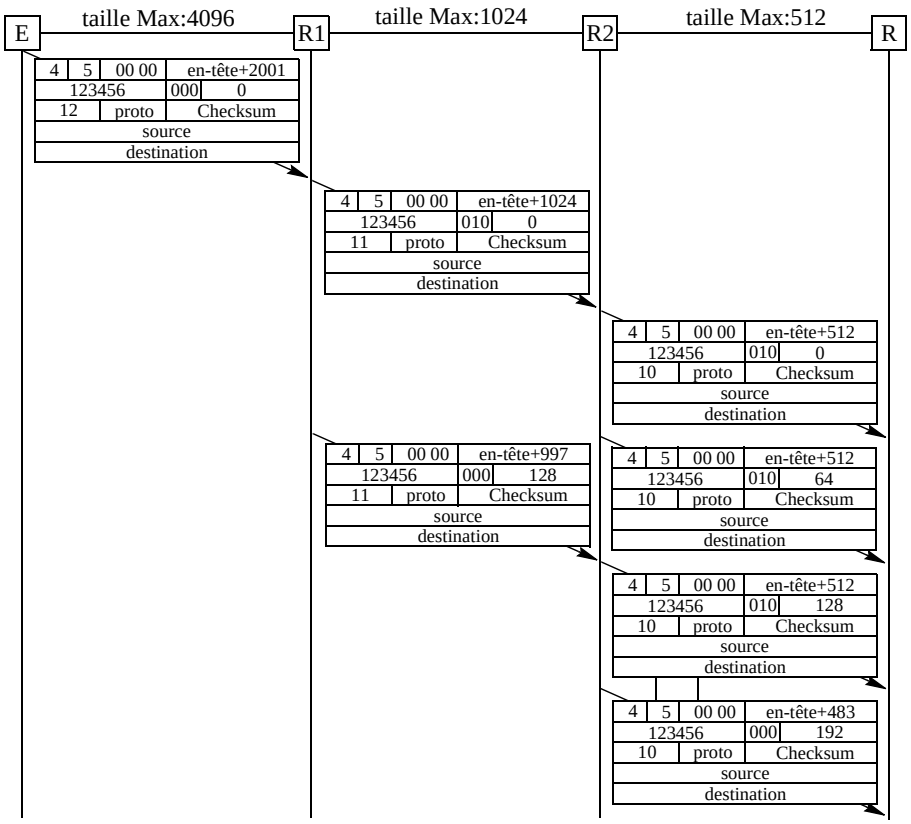

Figure 8.11. *Fragmentations d'un paquet*

— le troisième bit MF (*More Fragments*), s'il est à 0, indique qu'il s'agit du dernier fragment. S'il est à 1, le routeur a d'autres fragments du paquet originel à émettre.

Par défaut ces deux bits sont à 0. Ce qui signifie que le paquet peut être fragmenté et qu'un paquet unique est aussi un dernier fragment ;

— Le champ « place du fragment » sur 13 bits, indique la position du premier octet dans le paquet total (non fragmenté). Le premier fragment est à la place 0. Comme ce champ contient moins de bits que le champ longueur du fragment, la découpe des blocs, sauf celles du dernier fragment, ne peut se faire que par multiple de 8 octets. La valeur « place du fragment » doit donc être multipliée par 8 pour obtenir la position réelle du premier octet.

La figure 8.11. montre une fragmentation multiple d'un paquet traversant deux routeurs R1 et R2. Pour simplifier l'exemple, la taille maximale indiquée pour chaque

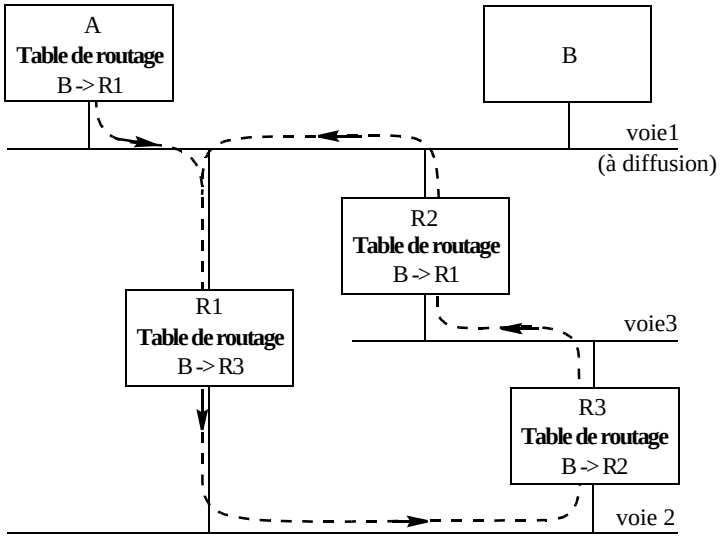

Figure 8.12. *Boucle dans les tables de routages*

réseau concerne uniquement les données. Le routeur R1 fragmente en deux parties le paquet de 2001 octets émis par E. Le premier fragment a une taille multiple de 8. Le second fragment contient le reste des données. Le champ début du fragment contient le nombre d'octets des paquets précédemment émis.

8.4.1.7. *Contrôle des boucles*

Le champ « durée de vie », sur 8 bits, indique soit le nombre de routeurs traversés, soit le temps passé en réassemblage. Il fournit une mesure du nombre de nœuds maximal que le paquet peut traverser dans le réseau ou du temps maximal que le paquet peut attendre avant réassemblage dans l'entité TCP destinataire. Si ce champ contient la valeur 0, alors le paquet doit être détruit. Cela permet d'éviter que les paquets bouclent trop longtemps dans le réseau ou qu'un fragment perdu empêche le réassemblage du paquet.

Chaque routeur qui traite un paquet décrémente sa valeur. Cela permet de détruire les paquets qui pourraient rester dans le réseau à la suite de boucles introduites dans le réseau du fait de la mise à jour des tables de routage (cf. figure 7.19. et paragraphe 7.2.3.). Il se comporte comme un compteur de routeur et donne le nombre maximal de routeurs qu'un paquet peut traverser. La figure 8.12. illustre ce problème. A la suite d'une erreur dans les informations de routage diffusées sur la voie1, les tables de la station A et du routeur R2 sont fausses. Ils croient que la station B se trouve sur la voie 2. La station A envoie son paquet au routeur R1, qui le retransmet au routeur R3. Ce dernier envoie le paquet au routeur R2, qui le donne au routeur R1. Le paquet tournera en permanence sur le réseau en consommant de la bande passante et des ressources financières si certaines des voies sont payantes.

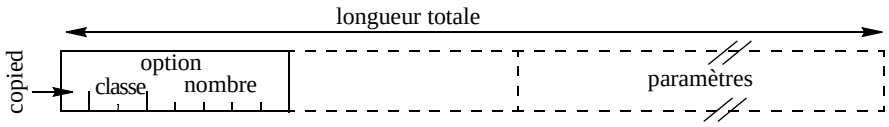

Figure 8.13. *Format d'un champ option*

Le site de réassemblage décrémente cette valeur (on retient celle du premier fragment reçu) chaque seconde jusqu'à ce que le paquet soit réassemblé et livré, ou que la valeur atteigne 0, auquel cas les fragments reçus sont jetés.

8.4.1.8. *SAP destinataire des paquets IP*

Le champ protocole, sur 8 bits, indique le protocole de la couche supérieure, autrement dit le SAP destinataire de ce paquet. Une liste des protocoles de niveau supérieur se trouve dans le RFC 1700. Les valeurs les plus couramment rencontrées sont :

```
ip        0 # internet protocol, pseudo protocol number
icmp      1 # internet control message protocol
tcp       6 # transmission control protocol
udp      17 # user datagram protocol
```

8.4.1.9. *Détection d'erreur sur l'en-tête IP*

Le checksum de l'en-tête est sur 16 bits : comme l'en-tête change en particulier à cause de la décrémentation du champ durée de vie, ce checksum doit être recalculé par chaque routeur avant retransmission. Le checksum est égal à la somme des mots de 16 bits de l'en-tête notés en complément à 1. Pendant le calcul, la valeur du checksum est mise à 0.

8.4.1.10. *Champs adresse de l'en-tête IP*

Les champs adresse de la source et destination sont sur 32 bits chacun : l'adresse source peut être de classe A, B ou C. L'adresse destination peut être de classe A, B, C ou D (paragraphe 6.2.1.).

8.4.1.11. *Options*

Le champ option est de longueur variable ou inexistant quand la longueur de l'en-tête vaut 5. Les options sont passées à la couche supérieure. Le type de l'option est codée sur un octet (cf. figure 8.13.). Le RFC 1700 donne la liste des options pour IP. Le lecteur trouvera dans [10] le détail des options et une description plus approfondie de IP et des protocoles associés.

8.5. Le protocole TCP (RFC 793)

TCP, *Transmission Control Protocol*, est un protocole de niveau transport qui permet un transfert fiable des données sur une connexion entre deux stations. Il est

identifié par la valeur 6 dans le champ protocole du paquet IP (cf. paragraphe 8.4.1.8.). TCP permet :

— un contrôle d'erreur sur les données transférées (données endommagées, perdues, dupliquées),

— une garantie de séquencement des données, même si la couche IP ne les délivre pas dans l'ordre,

— un contrôle de flux. Un mécanisme de fenêtre permet à TCP d'éviter d'envoyer des données à un récepteur qui ne possède pas d'espace en mémoire pour les recevoir,

— priorités : les données transmises dans un message peuvent être traitées plus ou moins prioritairement.

8.5.1. *Notion de port*

Au niveau 3, l'adressage (c'est-à-dire l'adresse IP) permet de désigner de manière unique une machine située n'importe où sur le réseau. Une fois la machine trouvée, il faut pouvoir déterminer l'application qui doit traiter les données. Cette désignation se fait par des SAP appelés « numéro de port ». Certaines applications, quelle que soit la machine, ont des numéros de port fixés et bien connus. Il s'agit en général des applications réparties connues de tous les systèmes. Le RFC 1700 donne la liste des numéros de port attribués aux applications universellement connues (cf. figure 8.14.). Les numéros de 0 à 1 023 sont réservés à ces points d'accès universel (*well known port*). Les numéros supérieurs à 1 024 peuvent être attribués à n'importe quelle application. Il est donc important de se référer à ce RFC lorsque l'on définit une nouvelle application. Pour les systèmes UNIX, le fichier /etc/services donne l'affectation des numéros des ports sur la machine.

8.5.2. *Format des messages TCP*

Le format des PDU (ou messages) est donné par la figure 8.15. :

— le port source et le port destination permettent de référencer les applications qui s'exécutent sur les machines locales et distantes. Ils construisent avec les adresses IP source et destination une référence unique (SAP) qui identifie la connexion parmi toutes les autres du même site ;

— le champ numéro de séquence : ce numéro indique le numéro du premier octet transmis dans le segment. Il est fonctionnellement équivalent au champ N_S que nous avons vu avec le protocole LAP B. Un compteur V_S est associé dans le contexte de la connexion. Si le bit SYN est positionné, le numéro de séquence contient une valeur (ISN : *Initial Sequence Number*) initialisée par un mécanisme d'horloge. Le premier octet transmis sera égal au numéro de séquence ISN+1 ;

Nom	Valeur	Protocole	Commentaire
ftp-data	20	tcp	Utilisé par ftp pour transférer des données (fichiers, répertoires, ...)
ftp	21	tcp	Utilisé par ftp pour transmettre des ordres.
telnet	23	tcp	Terminal virtuel
smtp	25	tcp	Courrier électronique
tftp	69	udp	Transfert de fichiers de configuration.
finger	79	tcp	Information sur l'utilisateur
sunrpc	111	udp, tcp	Montage de fichiers à distance
login	513	tcp	rlogin
talk	517	udp	

Figure 8.14. *Affectation des numéros de ports*

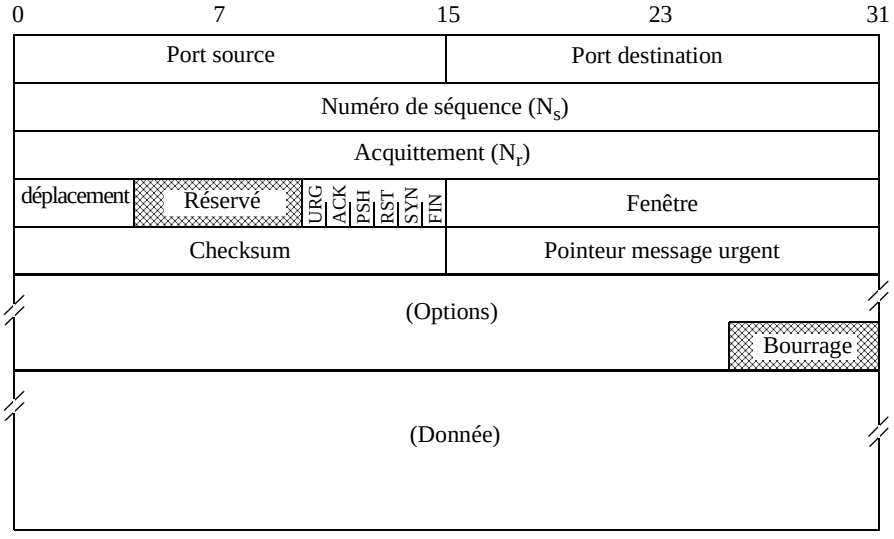

Figure 8.15. *Format d'un PDU TCP*

— le champ acquittement : ce numéro contient le numéro de séquence du prochain octet attendu par l'émetteur de ce message. Il est fonctionnellement équivalent au champ N_r que nous avons vu pour LAP B. Un compteur V_r est associé dans le contexte de la connexion. Une fois que la connexion est établie, cette valeur est toujours transmise ;

— le champ déplacement (4 bits) indique la taille de l'en-tête protocolaire en mots de 32 bits. Sa valeur est 5 pendant les échanges de donnée, et éventuellement 6 lors de l'établissement de la connexion ;

— les bits réservés (6 bits) : réservé pour un usage futur ;

— les bits de contrôle : ces bits permettent de définir la fonction des messages ainsi que la validité de certains champs :

- URG : la valeur du champ « pointeur message urgent » est significatif,
- ACK : la valeur du champ « acquittement » peut être prise en compte,
- PSH (*Push*) : les données reçues doivent être immédiatement transmises à la couche supérieure,
- RST (*Reset*) : réinitialisation de la connexion,
- SYN : ouverture de la connexion (ou resynchronisation des membres).
- FIN : fin de connexion (plus de données à émettre) ;

— le champ fenêtre : nombre d'octets que le récepteur peut accepter ;

— le champ checksum de l'en-tête et du message ;

— le champ pointeur message urgent : ce pointeur indique les octets qui doivent être traités en priorité.

— le champ option : ce champ permet de définir, par exemple, la taille maximale d'un segment (MSS, *Maximum Segment Size*), autrement dit la taille maximale d'un T PDU.

8.5.3. *Ouverture d'une connexion*

Les deux sites qui vont dialoguer ont un rôle dissymétrique. Dans la suite de ce chapitre, nous appellerons client la station qui demande l'ouverture de la connexion et serveur la station qui répond à cette demande. Nous étudierons au chapitre 9 les primitives d'établissement d'une connexion sous UNIX. Avant toute ouverture de connexion, le serveur doit avoir autorisé l'ouverture passive de la connexion. Comme pour LAP B, l'entité protocolaire maintient un contexte de connexion dont les variables lui permettent de gérer le protocole.

Le client demande une ouverture active de la connexion. Ce qui se traduit par l'émission d'un PDU avec le bit SYN à 1 sur le réseau avec le numéro de port serveur destinataire et un numéro de port client. La valeur ISN du champ « numéro de séquence N_s » est produite aléatoirement. Cette valeur incrémentée de 1 est conservée dans le compteur local V_s. La valeur du champ acquittement N_r est nulle car non définie.

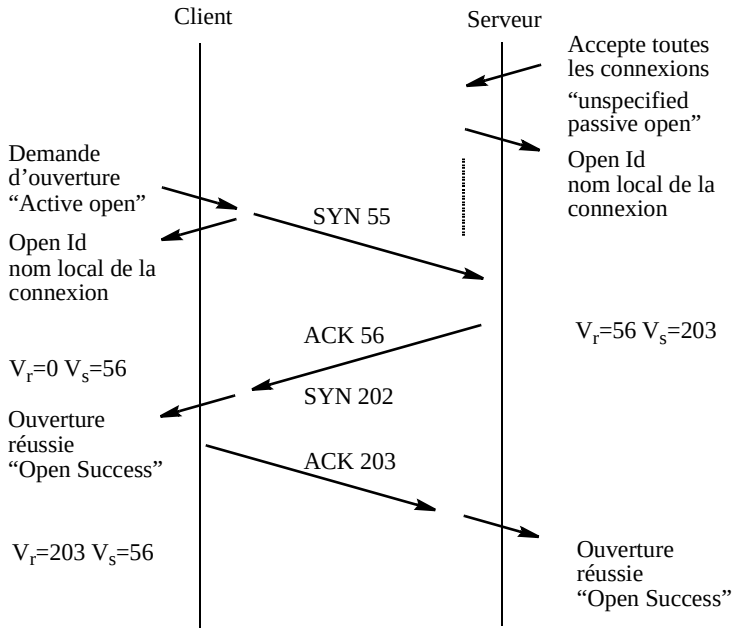

Figure 8.16. *Exemple de connexion réussie*

L’entité TCP homologue accepte la connexion si préalablement une application serveur lui a signalé qu’elle acceptait les connexions sur le port destinataire correspondant. Dans ce cas, elle renvoie un message avec les bits SYN à un et ACK à un. La valeur ISN contenue dans le champ « numéro de séquence N_s » est produite aléatoirement et conservée incrémentée de 1 dans le compteur V_s. La valeur du champ acquittement N_r est égale à $V_r = N_s+1$. Le contexte de la connexion est complètement établi. La connexion est immédiatement ouverte dans le sens serveur vers client.

Le client affecte à son compteur V_r la valeur du champ N_s + 1 du message reçu et envoie un PDU ACK vers le serveur pour protéger l’établissement de connexion contre les pertes de message. La valeur du champ « numéro de séquence N_s » est V_s. La valeur du champ acquittement N_r est égale à N_s+1. La figure 8.16. montre les échanges de message lors d’une ouverture de connexion TCP réussie.

Les numéros de séquence désignent le numéro du prochain octet attendu. Ce sera le numéro du premier octet du prochain PDU de donnée reçu dans le flot. Les numéros de séquence initiaux sont produits à partir d’un système d’horloges locales logiques. Un compteur est incrémenté toutes les 4 ms. La taille du champ séquence étant de 32 bits, la durée avant que le compteur ne revienne à zéro est à peu près de 4 heures 35 minutes. Ainsi, une connexion identifiée par le même couple émetteur et récepteur n’a pas de grand risque de retomber sur un message d’une connexion antérieure ayant les mêmes valeurs.

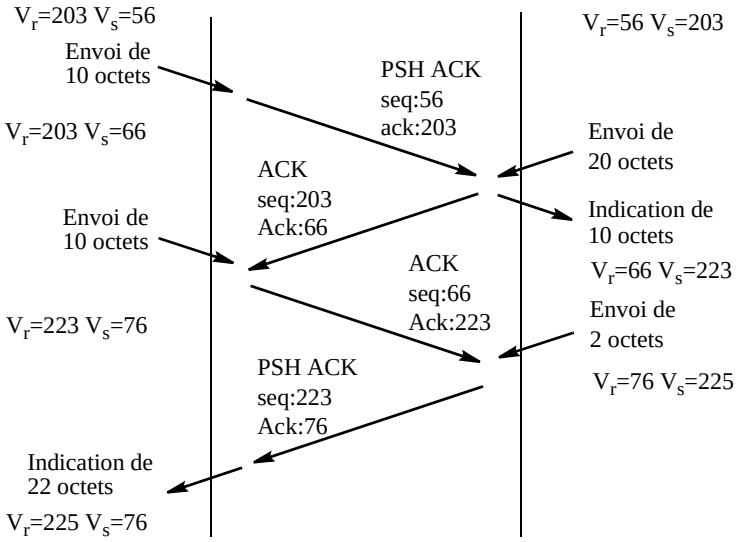

Figure 8.17. *Exemple de transfert de données*

8.5.4. *Transfert de données*

Contrairement aux protocoles définis dans l'OSI (X25.3) où l'unité est le paquet, le protocole TCP comptabilise les octets transmis. C'est-à-dire que le compteur V_s est augmenté du nombre d'octets transmis par le paquet émis. Lors de la réception sans erreur d'un paquet en séquence ($V_r = N_s$), le compteur V_r est augmenté du nombre d'octets reçus dans le paquet. L'exemple de la figure 8.17. montre l'enchaînement des paquets pour l'émission de données sur le réseau. Le champ « seq » indique la place du premier octet de données du paquet dans le flux de données et « ack » le prochain caractère attendu par l'émetteur du paquet.

Quand le bit PSH n'est pas positionné, le récepteur place les données dans un tampon et ne transmet les données à la couche supérieure que quand celui-ci est rempli. Par contre, quand le bit PSH est positionné, les données sont dès réception transmises à la couche supérieure.

La méthode pour gérer les erreurs de transmission, c'est-à-dire des pertes de paquets, est assez particulière en comparaison de celle utilisée par l'ISO. TCP n'utilise pas d'acquittement négatif pour signaler les erreurs. Celles-ci sont détectées par l'absence d'acquittement positif lors de l'expiration d'un timer. Toutes les données reçues hors séquence par le destinataire seront mémorisées.

TCP est un protocole orienté flux de données et non paquets. Aussi, lors d'une retransmission, les paquets initiaux ne sont pas retransmis à l'identique : TCP regroupe l'information afin d'améliorer le rendement.

La figure 8.18. donne le chronogramme d'un échange avec perte de paquets sur un réseau local. Les délais d'acheminement sont courts, ce qui dans cet exemple a empêché d'avoir plusieurs segments non acquittés. Sur le schéma ne figurent que le champ identificateur du paquet IP, les champs séquence et acquittement, ainsi que les drapeaux[1] des paquets TCP et la taille des données transportées.

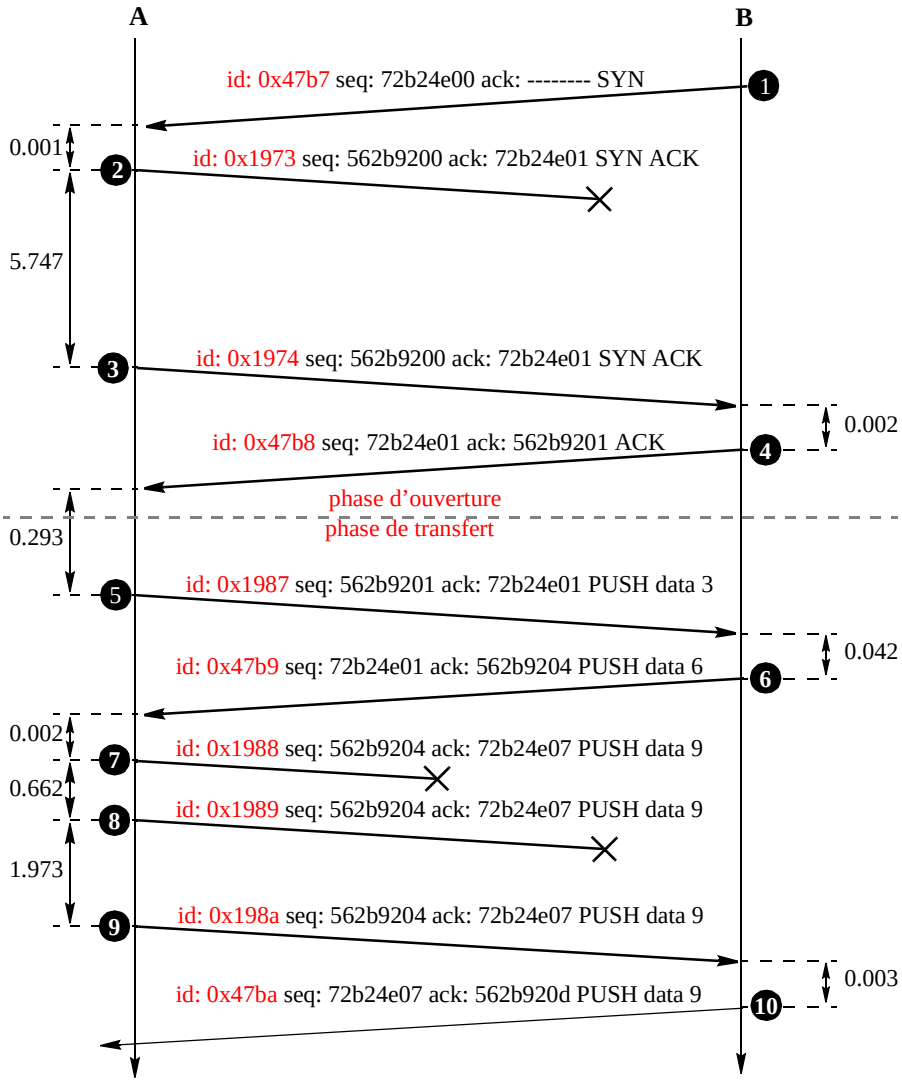

Figure 8.18. *Echange de données avec erreurs*

1. Le drapeau ACK étant toujours présent après la phase d'ouverture, il a été omis.

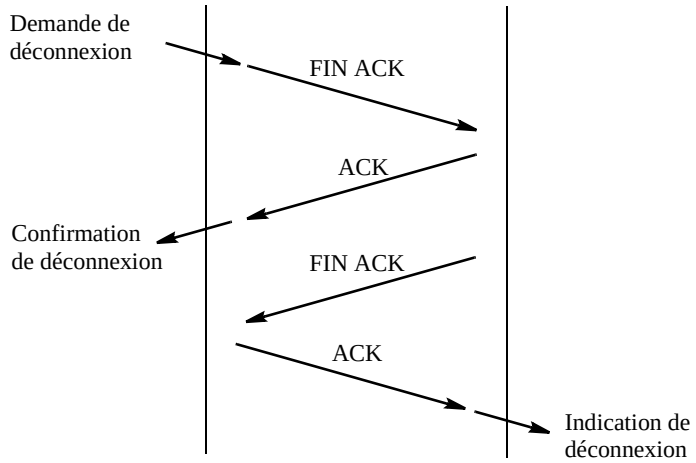

Figure 8.19. *Terminaison d'une connexion TCP*

La connexion est ouverte à l'initiative de la machine B (paquet 1). La machine A accepte la connexion et émet le paquet 2, qui est perdu. La machine A ne reçoit pas de réponse de B puisque son acceptation d'ouverture ne lui est pas parvenue. Au bout de 5,747 secondes, elle réémet la trame (paquet 3). Au niveau de IP, le numéro de séquence est différent du paquet 2 puisqu'il s'agit d'un nouveau paquet. Par contre, les champs séquence et acquittement sont identiques. La machine B répond (paquet 4) : la connexion est ouverte.

La machine A émet 3 octets de données (paquet 5) qui sont acquittés par le paquet 6. Le champ acquittement est incrémenté du nombre d'octets transmis.

La machine A émet 9 octets de données, mais le paquet 7 est perdu par le réseau. Comme la machine A ne reçoit aucun acquittement pour ce paquet, elle le réémet (paquet 8), mais celui-ci se perd au son tour. Finalement, la deuxième tentative de retransmission (paquet 9) arrivera à la machine B, qui acquittera les données grâce au paquet 10.

8.5.5. *Fermeture d'une connexion*

La fermeture d'une connexion (cf. figure 8.19.) se fait quand le récepteur reçoit une trame dont le bit FIN est positionné à 1.

8.6. Le protocole UDP (RFC 768)

UDP, *User Datagram Protocol*, permet à une application d'envoyer des messages vers une autre avec un minimum de fonctionnalités (pas de garantie d'arrivée ni de contrôle de séquencement). Il apporte une seule fonctionnalité supplémentaire par

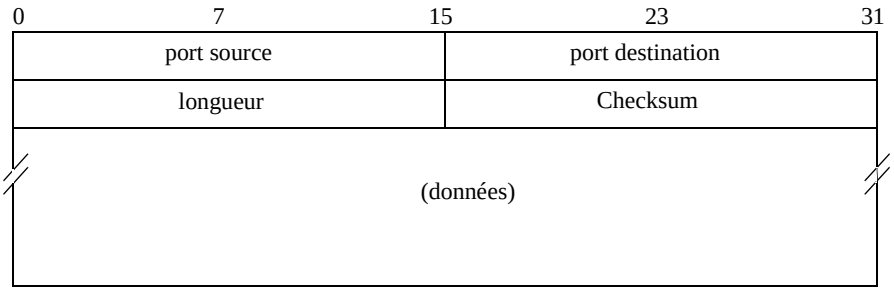

Figure 8.20. *Format d'un message UDP*

rapport à IP, qui est d'utiliser les numéros de port, et de ce fait de se mettre au même niveau que TCP. Un message UDP est désigné dans un paquet IP par une valeur du champ protocole égale à 0 x 11 (17_{10}) (cf. paragraphe 8.4.1.8.).

Le format d'une trame UDP est donné figure 8.20 :

— le port source et le port destination permettent de référencer les applications qui s'exécutent sur les machines locale et distante,

— la longueur indique la longueur totale du message en octets (données + en-tête),

— le checksum de l'en-tête est calculé comme pour les paquets IP. Un checksum à 0 indique qu'il n'est pas utilisé.

La figure 8.21. donne un exemple de PDU (paquet) UDP encapsulé dans un paquet IP.

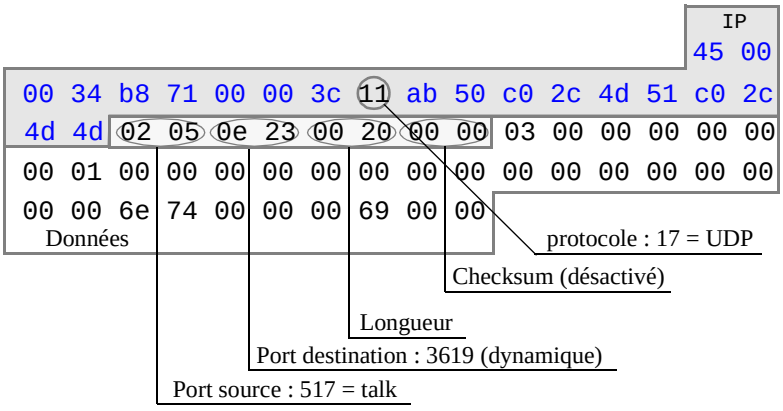

Figure 8.21. *Exemple de PDU (paquet) UDP encapsulé dans un paquet IP*

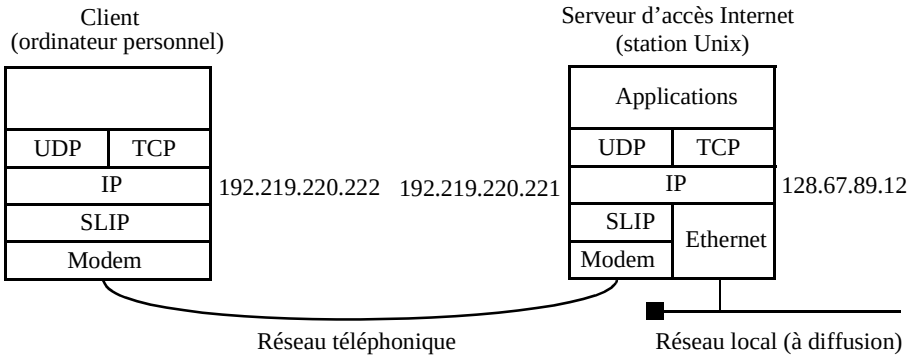

Figure 8.22. *Architecture d'une liaison SLIP*

8.7. Intégration de SLIP-IP et TCP/UDP

Une station mettant en œuvre SLIP et au moins une autre voie se comporte comme un routeur. On appelle interface le point d'accès à une voie (ensemble logiciel plus matériel). Un sous-réseau IP correspond à la liaison SLIP (cf. figure 8.22.). Le serveur possède une adresse IP sur le réseau local et une adresse sur la liaison série.

La table de routage chez le serveur d'accès Internet permet d'intégrer l'ordinateur personnel dans le réseau. L'ordinateur personnel n'a pas besoin d'une table de routage complexe : tout son trafic est routé vers le serveur d'accès Internet par la voie série SLIP. Le contenu de la table du serveur (routeur) relativement à la figure 8.22. est donné figure 8.23.

Pour aller	Passer par
Sur la machine 192.219.220.222	L'interface locale connectée au réseau téléphonique 192.219.220.221
Sur le réseau 128.67.89.0	L'interface locale connectée au réseau local 128.67.89.12
Ailleurs (routage par defaut)	Le routeur 128.67.89.1

Figure 8.23. *Table de routage du serveur*

La première ligne indique que, pour joindre la machine 192.219.220.222, il faut utiliser l'interface correspondant à l'adresse Internet 192.219.220.221. La deuxième ligne décrit l'attachement de la station au réseau local ; pour joindre les machines situées sur le réseau, 128.67.89.0, correspondant à l'adresse 128.67.89.12. La troisième entrée donne le routage par défaut. Quand l'adresse ne correspond à aucune

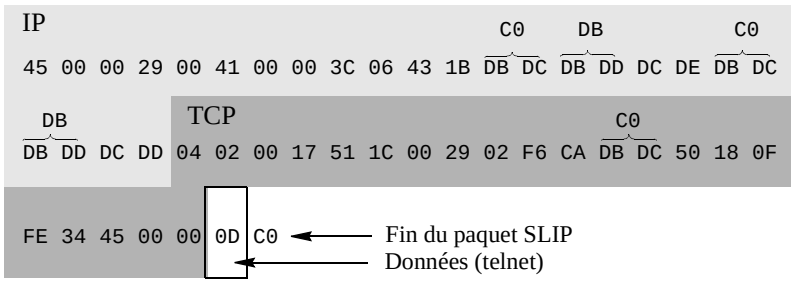

Figure 8.25. *Paquet sur une liaison SLIP*

des adresses précédemment définies, le paquet doit être envoyé au routeur 128.67.89.1. En consultant sa table de routage, l'équipement voit que l'adresse 128.67.89.1 est accessible par l'interface connectée au réseau local.

La configuration chez le client est plus simple. Avant la connexion, le client doit configurer son ordinateur avec son adresse Internet et le masque[1] de sous-réseau associé, donner l'adresse du routeur par défaut (il s'agit du serveur auquel il est raccordé) et l'adresse d'un serveur de nom. La figure 8.24. montre les paramètres de configuration de l'interface série à l'aide du logiciel Trumpet.

8.7.1. *Exemple*

La figure 8.25. donne un exemple d'un paquet échangé sur une liaison SLIP par une application telnet entre deux machines ayant les adresses 192.219.220.221 et 192.219.220.222.

Le lecteur remarquera les séquences esc (0xDB) et esc_end (0xDC) pour coder la valeur oxC0 et les séquences esc (0xDB) et esc_esc (0xDD) pour coder le caractère esc (0xDB).

8.7.2. *Limitations*

Une des limitations de SLIP est contenue dans son nom. Le fait que le paquet soit directement émis sur la liaison série ne permet pas de transporter un autre protocole : le multiplexage n'est pas possible sur la voie série. Il aurait fallu au moins un octet supplémentaire pour autoriser le multiplexage de différents protocoles (IPX, Appletalk...).

SLIP suppose que la liaison série est exempte d'erreur. Il n'existe, à son niveau protocolaire, aucun mécanisme qui permette de détecter et a fortiori de récupérer une

1. Le masque permet de définir les bits de l'adresse communs à toutes les stations d'un sous-réseau.

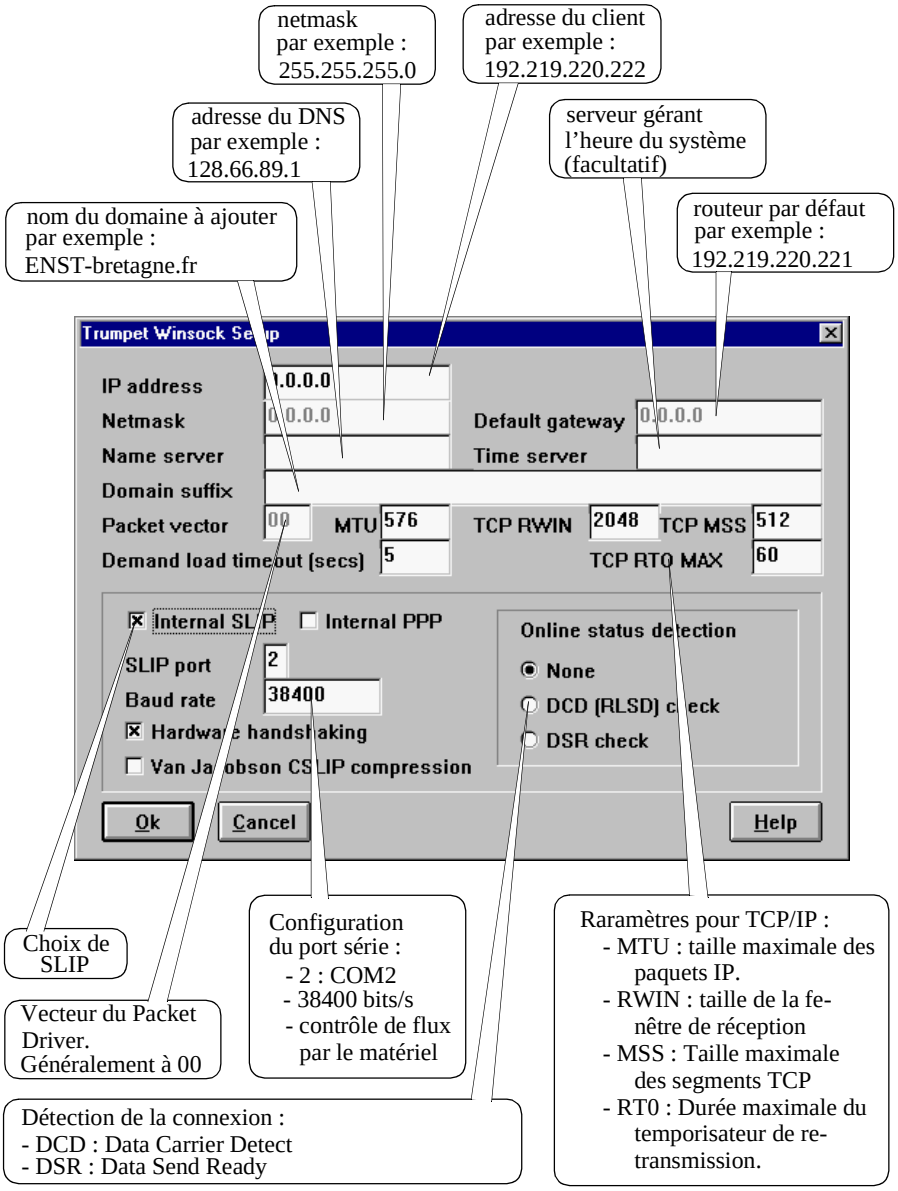

Figure 8.24. *Exemple de configuration avec l'application Trumpet pour PC*

erreur de transmission. Celles-ci peuvent être détectées au niveau IP par le checksum et doivent être corrigées au niveau 4 avec TCP ou avec UDP au niveau applicatif.

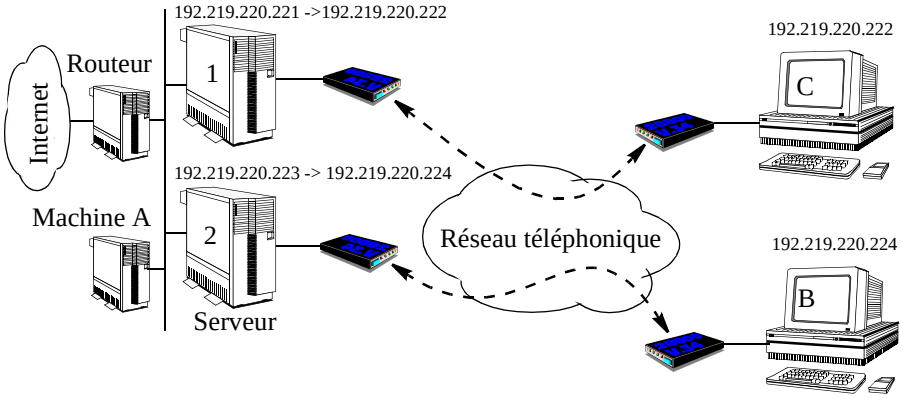

Figure 8.26. *Affectation dynamique des adresses IP*

SLIP suppose aussi que la liaison est neutre, c'est-à-dire que les ETCD ne vont ni interpréter les caractères transmis ni en introduire.

Dans le cas d'applications interactives comme telnet, la ligne sera utilisée principalement pour transmettre des en-têtes. Sur la figure 8.25, la trame transmise a une taille de 41 octets mais ne contient qu'un seul octet de données utile. Si la liaison série offre un faible débit, cela se ressentira au niveau des performances.

L'exemple figure 8.26. illustre d'autres problèmes rencontrés avec les liaisons SLIP. Une entreprise propose à ses employés un accès à son réseau Internet en utilisant un modem connecté sur la machine 1. Chaque employé possède un ordinateur configuré avec la même adresse IP attendue par le serveur (voir la configuration des tables de routage pour SLIP sur la figure 8.23.). La duplication de l'adresse IP sur plusieurs ordinateurs personnels ne pose pas de problème tant qu'un seul peut se connecter à la fois.

Vu le succès du service, l'entreprise décide d'ouvrir un second accès. Pour le rendre transparent, elle rajoute une ligne ayant le même numéro téléphonique. Les utilisateurs ignorent, lorsqu'ils appellent, sur quel modem ils seront connectés. Les équipements distants ne peuvent pas avoir la même adresse IP, sinon le serveur ne saurait pas où envoyer les données.

Quand un employé se connecte au réseau, il devra configurer son ordinateur pour que son adresse soit adéquate et qu'il puisse ainsi la garder pendant la durée de la connexion.

Le dernier problème concerne le contrôle d'accès au réseau. SLIP ne permet pas de limiter l'accès des ordinateurs qui se connectent aux modems de l'entreprise. Les mots de passe empêchent un intrus de se connecter aux ordinateurs de la société, mais rien ne peut l'empêcher de se servir des modems pour se connecter au serveur et

ensuite d'appeler un autre équipement sur lequel il a un compte quelque part sur le réseau Internet.

Le protocole PPP *(Point to Point Protocol)* résout ces problèmes et est de plus en plus fréquemment utilisé en lieu et place de SLIP. PPP permet le multiplexage de plusieurs protocoles (IP, Appletalk, IPX...) sur la voie. Il permet en outre la mise en œuvre de mécanismes de sécurité par mot de passe et vérifie périodiquement (de manière transparente pour l'usager) l'authenticité de l'utilisateur. PPP utilise des techniques de compression de l'en-tête pour améliorer le rendement. Ces améliorations par rapport à SLIP se payent par une plus grande complexité du protocole et de la définition des paramètres de mise en œuvre.

8.8. Conclusion

Le lecteur aura remarqué que les principes du protocole TCP pour la correction d'erreur et le séquencement des messages sont très similaires à ceux étudiés avec LAP B. Le principe du mécanisme de comptage des informations envoyées est quasi universel pour les protocoles qui garantissent le séquencement. Les différences avec LAP B sont :

— LAP B ou X25 niveau 3 numérotent les paquets transmis, TCP numérote les octets transmis. Cela nécessite des compteurs plus larges : 6 bits pour LAP B ou X25 niveau 3 au lieu de 64 bits pour TCP, et entraîne un rendement moins bon.

— Une entité TCP récepteur qui détecte une rupture dans le séquencement des données lors d'une connexion ne peut pas envoyer un acquittement négatif. La retransmission se fera par déclenchement d'un réveil chez l'émetteur.

— TCP repose sur le protocole IP en mode datagramme. Lorsque la fragmentation est nécessaire, il faut utiliser un champ séquence afin d'identifier les paquets, un champ place du fragment et un bit MF afin d'indiquer si le fragment est le dernier pour prendre en compte les déséquencements. Du plus, si un fragment se perd, le paquet est entièrement perdu. Pour X25 niveau 3, seul un bit M est nécessaire.